AD

TECHNICAL REPORT
ECOM-00240-1, VOL. III

LIGHT TRANSPORT IN THE ATMOSPHERE

Volume III: Utilization Instructions for the LITE Codes

ANNUAL REPORT
1 August 1965 to 31 August 1966

By
D. G. COLLINS, M. B. WELLS,
and K. CUNNINGHAM

SEPTEMBER 1966

ECOM
UNITED STATES ARMY ELECTRONICS COMMAND • FORT MONMOUTH, N.J.
CONTRACT DA28-043 AMC-00240(E)
RADIATION RESEARCH ASSOCIATES, INC.
Fort Worth, Texas

Qualified requestors may obtain copies of this report from DDC.
Distribution of this report is unlimited.

TECHNICAL REPORT ECOM-00240-1, VOL. III
RRA-T63 3

SEPTEMBER 1966

LIGHT TRANSPORT IN THE ATMOSPHERE

VOLUME III: UTILIZATION INSTRUCTIONS FOR THE LITE CODES

Annual Report
1 August 1965 to 31 August 1966

Contract No. DA 28-043 AMC-00240(E)

Prepared by

D. G. Collins, M. B. Wells, and K. Cunningham

RADIATION RESEARCH ASSOCIATES, INC.
Fort Worth, Texas

for
U. S. Army Electronics Command, Fort Monmouth, New Jersey

ABSTRACT

This is the third of three volumes. Volumes I and II contain other aspects of the study: descriptions of the RRA-42 and RRA-45 codes and their applications to the calculation of aerosol attenuation coefficients and the applications of the LITE codes to analysis of experimental data.

The Monte Carlo procedures designated as LITE-I and LITE-II were developed during a previous contract period for use in studying the transport of light through the earth's atmosphere under various environmental conditions. These procedures have been modified to expand their application to a broader range of physical problems. LITE-I treats monochromatic light emitted from a point source, and LITE-II treats monochromatic plane sources of light. The codes have been written in both ALGOL for the Burroughs B-5500 and FORTRAN-IV for other computers. The codes are sufficiently flexible to treat multiple scattering in an atmosphere in which the air density and the aerosol size distribution vary independently and arbitrarily with altitude. Provision for treating ground and cloud reflection with an albedo method is also available in the codes.

A machine procedure, designated as ACC, was developed for use in converting the scattered intensities computed by the LITE codes for a given ground albedo to scattered intensities for problems where only the magnitude of the ground albedo has changed.

Utilization instructions, input data formats, sample problems and the ALGOL listings of ACC and the improved versions of the LITE programs are given to aid those who wish to utilize the codes.

PREFACE

During the period 1 August 1965 to 31 August 1966 Monte Carlo studies were performed to determine light transport in the atmosphere under various environmental conditions. These studies consisted of 1) correlation analysis of light transport from a point isotropic source and a plane parallel source to determine the comparability of solar light transmission data and transmission properties for thermal radiation from nuclear weapons, 2) development of machine codes for calculation of phase functions and scattering and absorption coefficients for spherical-homogeneous aerosol particles with a complex index of refraction, 3) an analysis of experimental field data on light transmission, 4) parametric studies to determine the specific influence of ground and cloud albedo, cloud height and aerosol number density and particle-size distribution on the transport of light in the atmosphere, 5) modifications to the LITE codes to increase their application to a wider range of atmospheric transport problems and 6) the development of a machine program for use in converting the scattered intensities computed by the LITE codes for a given ground albedo to data giving scattered intensities and scattered fluxes for other ground albedos. The results of these studies are presented in this report, which is divided into three volumes. The first volume describes the results of items 1, 3, and 4 outlined above. The second volume describes the machine programs developed for use in calculation of aerosol cross sections. The third volume contains utilization instructions for the modified versions of the LITE codes and for the code developed to convert the LITE results to data giving scattered intensities and fluxes for other ground albedos.

FOREWORD

The authors wish to express their appreciation to Henrietta Hendrickson and Hemma Francis of Oak Ridge National Laboratory who aided in the check out and running of test problems of the FORTRAN-IV version of the LITE codes. They also wish to acknowledge the assistance of Leon Leskowitz of the U. S. Army Electronics Laboratory in translating the FORTRAN-IV version of the LITE code to ALGOL language and in scheduling the LITE problems run on the B-5500 computer. Technical Monitors of the work described in this report were I. Cantor of the Atmospheric Sciences Laboratory, USAECOM, Fort Monmouth, New Jersey and R. W. Fenn of the Air Force Cambridge Research Laboratories, Bedford, Massachusetts.

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

LIST OF TABLES

LIST OF FIGURES

I. INTRODUCTION

The two Monte Carlo programs, LITE-I and LITE-II, which were developed during a previous contract period (Ref. 1) for use in studying the effects of atmospheric and terrain conditions on the transmission of visible light in the earth's atmosphere have been converted to FORTRAN-IV. The FORTRAN-IV versions have also been translated to the ALGOL language for execution on the Burrough's B-5500 computer. The two programs, LITE-I for point sources of light and LITE-II for plane sources, have been utilized in studies of light transmission over the past year (Ref.2). During this period several minor modifications were made in the programs to expand the application of the programs to cover a broader range of physical problems.

An auxiliary program was written to convert the scattered intensities computed by the LITE programs for a given atmospheric condition and ground albedo to data for problems in which all the input parameters are unchanged except the albedo for the first reflection surface. This program designated ACC, Albedo Conversion Code, will also calculate the light current through a plane normal to either one of the three coordinate axes used in defining the scattered angular intensities computed by the LITE codes.

The modifications made to the LITE codes during the contract period are discussed in Section II. The ACC is discussed in Section III. Sections IV and V contain the utilization instructions and sample problems for the LITE programs and Sections VI and VII give the utilization instructions and a sample problem for the ACC. The ALGOL listings of LITE-I and LITE-II and ACC are given in Section VIII.

II. LITE CODES

A discussion of the modifications made to the LITE programs is preceded by a brief description of the methods utilized in the program. For a more detailed description of methods the reader is referred to Ref. 1.

2.1 Method Description

The LITE-I and LITE-II Monte Carlo programs were designed so that atmospheres could be described in which the air density and aerosol content both vary independently and arbitrarily with altitude. The first of these programs was developed to study the transport of monochromatic light emitted isotropically or with an arbitrary polar angle distribution by a point source located in an air-ground geometry This program has been designated as the LITE-I code. The second program, LITE-II, was developed to study the transport of monochromatic light emitted from a plane source with an arbitrary polar angle distribution located at the top of the atmosphere or within the atmosphere.

Routines are available in the programs for treating both Rayleigh and aerosol scattering events. An intermixture of the two events is possible or the atmosphere may be considered to be either a Rayleigh or an aerosol atmosphere. The atmosphere may be subdivided into plane slab regions and a different aerosol phase function input for each region. Thus, the scattering properties of cloudy and non-cloudy atmospheres may be defined with a high degree of accuracy.

Albedo techniques are incorporated in both programs to treat both ground and cloud reflection; however, either the ground or cloud regions

may be treated as regions in which both scattering and absorption can occur, if desired.

2.2 Modifications

Four significant modifications have been incorporated into the LITE programs during the past year. The first of these was made only in LITE-I. Originally LITE-I was designated to treat only light radiation emitted uniformly in all azimuthal directions by a point source with an arbitrary input polar angle distribution. This restriction on the source description ruled out any use of the code in studying atmospheric scattering of light from line beam sources such as lasers unless the beam was directed vertically. The restriction on the source description was removed by providing for the input of an arbitrary source azimuthal angle distribution from which to select the azimuthal directions of the source photons. By defining the source angle distributions to have values only for polar angles θ (see Fig. 1) in the interval between ϕ_1 and ϕ_2, then the source will emit radiation only in the solid angle defined by

$$SA = (\phi_2 - \phi_1)\ (\cos\theta_1 - \cos\theta_2) \quad .$$

The statistical fluctuation of the LITE-I results for problems having uniform azimuthal emission of light from a point source indicated the need for biasing the sample from the azimuthal angle distribution. A biasing scheme was developed which favors those azimuthal angles near zero degrees and the scheme was incorporated into LITE-I. The biasing scheme allows one to sample azimuthal angles from the density function

$$\frac{A\ e^{-A\phi}}{1 - e^{-A\phi max}}\ d\phi \qquad (1)$$

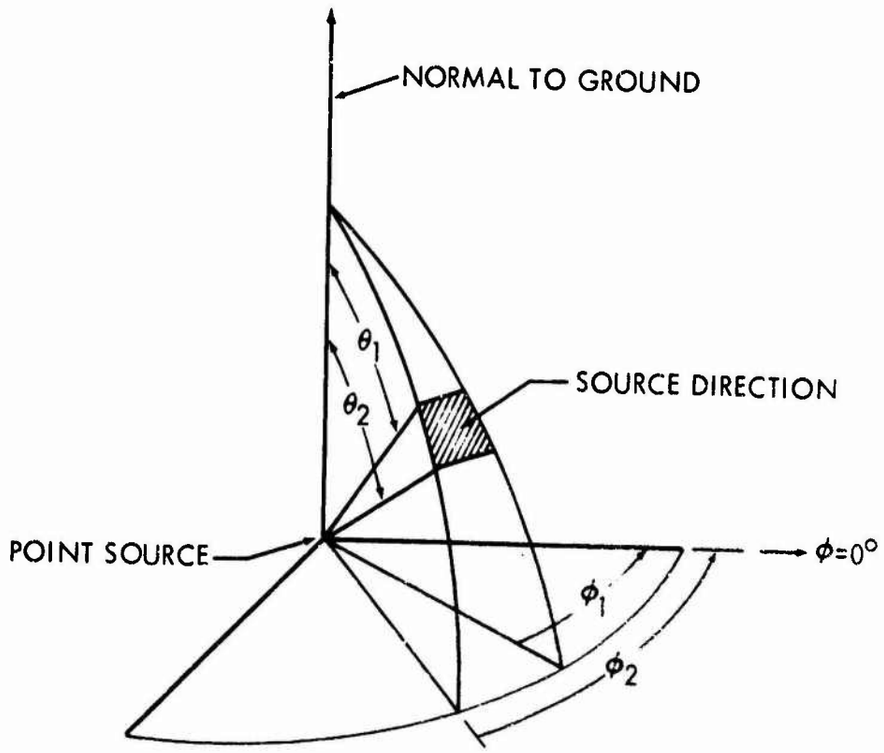


Fig. 1. Definition of Source Angles for LITE-I

where ϕ_{max} is equal to π or the largest angle used to describe the true azimuthal angle density function if that angle is less than π, and A is an input parameter.

The most effective use of the biasing scheme is accomplished when the azimuthal angles of all receiver positions are zero. The value input for the variable A determines the extent to which biasing is applied. If a negative or zero value is input for A, no biasing is applied and the source azimuthal angles are chosen directly from the cumulative probability table input to define the true azimuthal angle distribution.

If the largest angle used in describing the true azimuthal angle density function is greater than π, the code used the density function (1) with $\phi_{max} = \pi$. Then a random number is tested against 0.5 and if it is less than 0.5, the azimuthal angle selected, $\phi=\phi'$, is unchanged. However, if the random number is greater than 0.5, the azimuthal angle is taken to be

$$\phi = 2\pi - \phi'$$

where ϕ' is the angle chosen from the density function given in equation (1).

To correct for the bias introduced in the particle weight when azimuthal angles are sampled from equation (1), the particle weight is multiplied by the factor

$$((1-e^{-A\phi_{max}})/Ae^{-A\phi})p(\phi)$$

where $p(\phi)$ is the true probability density function evaluated at ϕ, the azimuthal chosen.

A word of caution should be given to those utilizing the LITE-I code. When A=0, the values input for PAZA(I) should describe the unbiased cumulative

azimuthal angular distribution, but when A>0, the values input for PAZA(I) should describe the unbiased non-accumulative azimuthal density function.

To aid the user in the selection of a value for the biasing parameter A, examples of cumulative distributions for various values of A are given in Figure 2. The probability

$$P(\phi') = \int_0^{\phi'} \frac{Ae^{-A\phi}d\phi}{1-e^{-A\pi}}$$

that the source particle's azimuthal direction will be between 0 and ϕ', is plotted versus ϕ' for several values of A. Note that for values of A near 0, the biased distribution is almost isotropic; but as A is increased, the biased distribution becomes more peaked in the forward direction. For A = 0.5, half of the particles are emitted with azimuthal angles between 0 and 58° and 68.5% of the particles have azimuthal angles less than 90°. When A is increased to 1.0, half of the particles are emitted within the first 37° and 82.5% have azimuthal angles less than 90°.

A second modification was made to both LITE-I and LITE-II to provide for an albedo which is dependent upon the angle of incidence. Several problems run with the LITE-II code for different angles of incidence upon a thick cloud indicated that the reflected distribution resembled a cosine distribution for all angles of incidence, but the total flux reflected was dependent upon the angle of incidence. It was also determined that the total number of particles reflected per particle incident at angle θ could be fitted with the expression

$$\text{ALBEDO} = C_1 + C_2 \cos\theta \qquad (2)$$

where C_1 and C_2 are constants and θ is the angle of incidence measured from the normal to the reflection surface. The expression (2) was incorporated

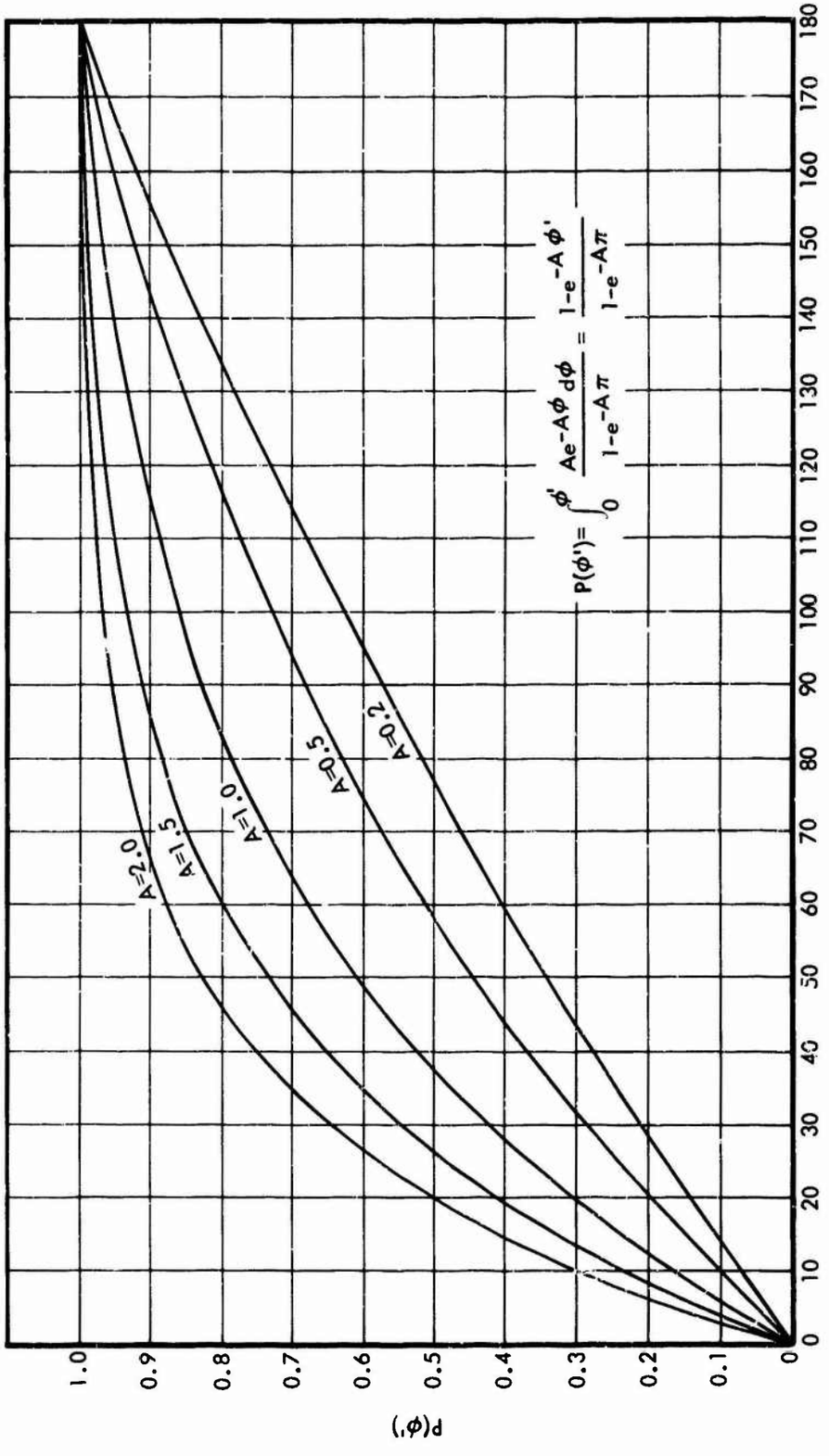


Fig. 2. Biased Cumulative Azimuthal Angular Distribution as a Function of Input Parameter A

in the LITE codes to allow the albedo to vary with incident angle. Previously the albedo had been defined with a single constant, ALBEDO = C_1.

The third modification made to the LITE codes was the addition of instructions to print the azimuthal angle dependence of the scattered intensity as well as the polar angle dependence. For the LITE-II code the azimuthal angle dependence of the scattered intensity is defined in terms of a coordinate system (see Fig. 3) that has the polar axis pointing vertically and the X and Y axes in the horizontal plane. The positive X axis is defined as the zero azimuthal angle direction.

The polar angle distribution of the scattered intensity as computed by use of LITE-I is given in terms of a coordinate system that has the polar axis coincident with the source-receiver axis (see Fig. 4). Thus the X and Y axes lie in a plane normal to the source-receiver axis. The X axis which defines the zero azimuthal angle at the receiver is contained in the vertical plane containing the source and receiver points. For a source point located at a height HS on the vertical axis and a receiver located at the position RD, HD, ϕD, the sine and cosine of the angle between the source-receiver axis and the vertical axis are given by the equations

$$SID = \frac{RD}{SOD}$$

$$COD = \frac{HS-HD}{SOD}$$

where SOD is the distance between the source and receiver. For a collision at the location R2, H2, ϕ2, the cosine of the polar angle between the source-receiver axis and the line joining the collision and receiver points is given by the expression

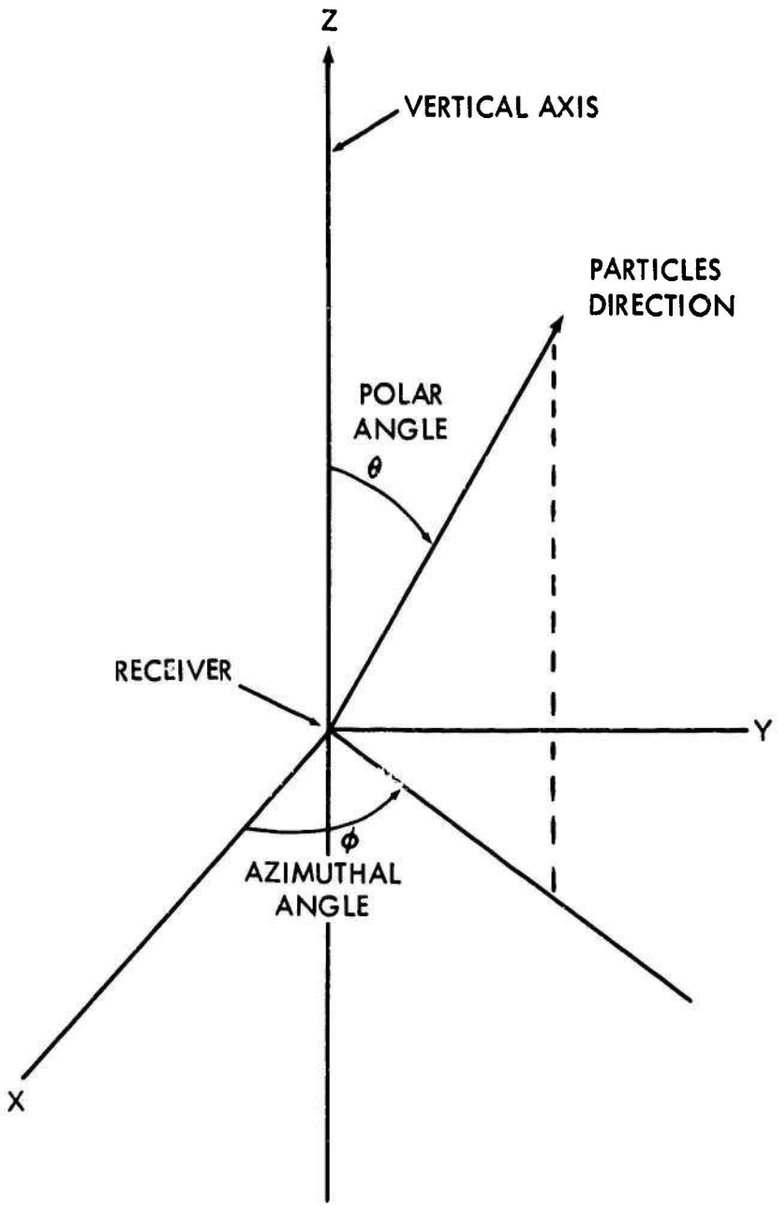


Fig. 3. Difinition of Print Angles for LITE-II

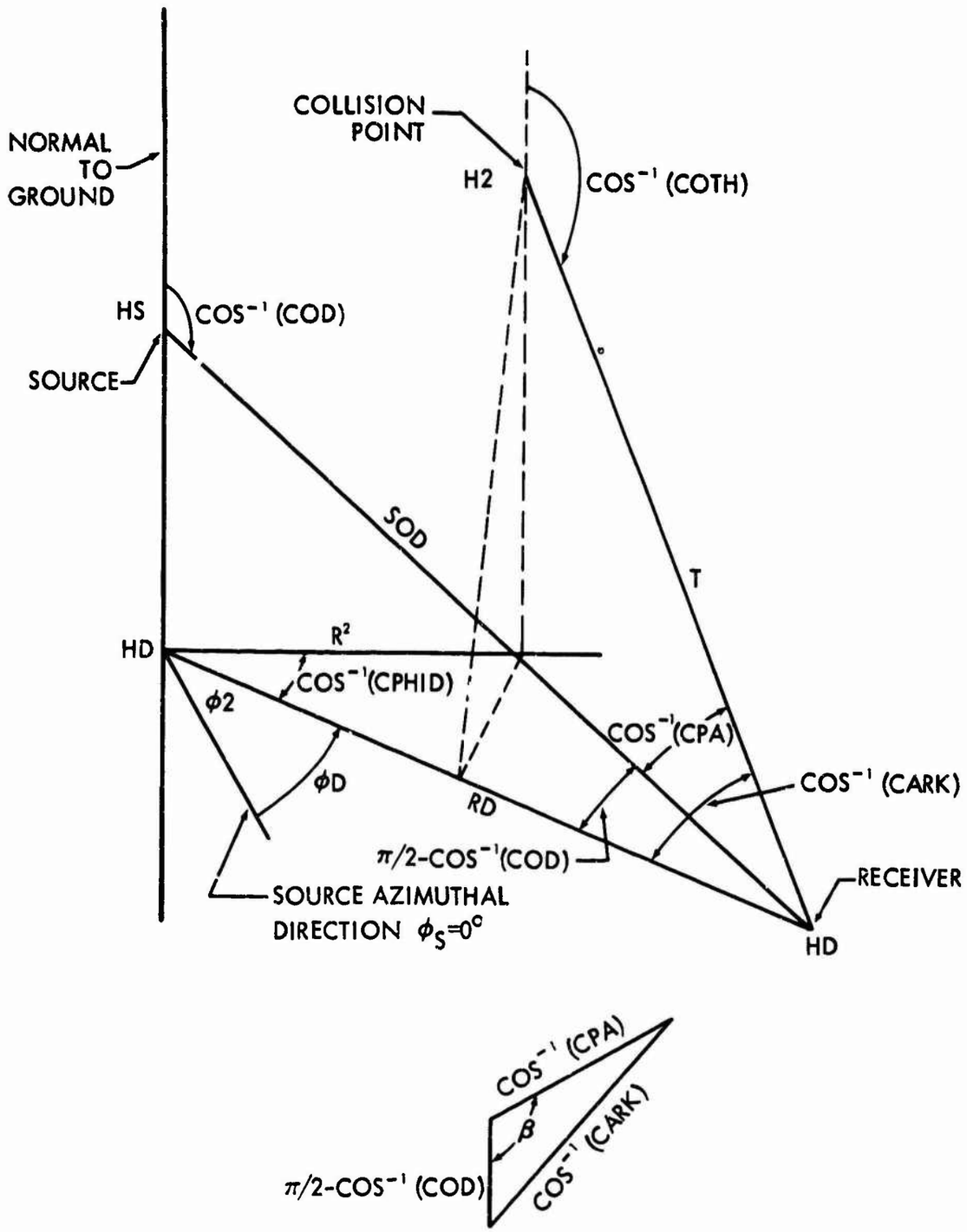


Fig. 4. Geometry for Calculation of the Receiver Polar and Azimuthal Angles in LITE-I

CPA = (CARK*SID)+(COD*COTH)

where COTH is the cosine of the angle between the vertical and the line joining the collision and receiver points and

CARK = (RD-(R2*CPHID))/T.

T is the distance between the collision and receiver points and CPHID is the cosine of the difference between the angles giving the azimuthal positions of the collision and receiver points. The projection of the line joining the collision and receiver points into a plane normal to the source-receiver axis makes an angle β with the X axis. Cosine β is given by the equation

$$\cos\beta = (CARK-(CPA*SID))/(SPA*COD),$$

where SPA is the sine of the polar angle between the source-receiver axis and the line joining the collision and receiver points. The angle β is the azimuthal angle used in the print format for LITE-I. When the altitude of the collision point, H2, is less than HS and the collision point lies on the plane containing the source-receiver axis and the vertical axis, then $\beta = 0°$.

A fourth modification made to the LITE codes provides for punching on cards the scattered intensities as a function of the polar and azimuthal angles and the number of times the particle has been reflected as well as printing them out. The punched output from the LITE codes may be used as a portion of the input to the Albedo Conversion Code which converts the output to apply to different albedos for the first reflection surface described in the LITE code input.

The modifications made in the LITE programs require that additional data be supplied as input to the two programs. The utilization instructions for the LITE codes have been revised to incorporate the additional input data and the revised utilization instructions are given in Section IV of this report.

The input and output of a sample problem for each program is given in Section V and the ALGOL listings of the improved versions of the LITE programs are given in Section VIII.

III. ALBEDO CONVERSION CODE

The LITE programs print out the scattered light intensity at a receiver as a function of the order of reflection from the first reflection surface given in the problem input. This data may be used to predict the scattered light intensity for problems in which all input parameters are unchanged except the magnitude of the albedo for the first reflection surface. An auxiliary program denoted as ACC, Albedo Conversion Code, has been written to convert the scattered intensities on the punched output from the LITE programs to data for different magnitudes of the albedo input for the first reflection surface. ACC may also be used to convert the LITE calculated scattered intensities to scattered currents across one of the three planes normal to the axes of the coordinate system used to reference the LITE problem print polar and azimuthal angles.

3.1 Methods Description

The punched card output of the LITE codes provides information on the amount of the scattered intensity $F(N,\theta(I),\phi(J),\alpha)$ that arrives at a given receiver with directions within the Ith polar angle interval and Jth azimuthal angle interval that results from the photons that have undergone exactly N reflections from a reflection surface having an albedo α. This data can be converted to give data for a reflection surface having an albedo α' by use of the equation

$$A(\theta(I),\phi(J),\alpha') = \sum_{N=0}^{Nmax} F(N,\theta(I),\phi(J),\alpha)\ (\alpha'/\alpha)^N \qquad (3)$$

where $A(\theta(I),\phi(J),\alpha')$ is the scattered intensity for albedo α' that is contained in the Ith polar angle interval and the Jth azimuthal angle interval,

NMAX is the maximum number of reflections for which the scattered intensities were originally computed. The lower bounds, CTHETA(2) through CTHETA (I_{max}) of the polar angle intervals are given in the LITE code punched output. CTHETA(1) is not shown in the LITE punched output, but is taken in ACC to be 1.0. The azimuthal interval, STER, is an input parameter for the ACC code.

In order to calculate the photon current (flux) through a plane normal to the polar axis, the equation for $CNP(\theta(I),\phi(J),\alpha')$ is

$$CNP(\theta(I),\phi(J),\alpha') = \sum_{N=0}^{NMAX} (\alpha'/\alpha)^N(M)\ F(N,\theta(I),\phi(J),\alpha)$$

where M is given by

$$M = |(\cos\theta(I) + \cos\theta(I+1))/2| \ .$$

When calculating the photon current through a plane containing the polar axis and a normal to the zero azimuthal axis, the equation for $CNA(\theta(I),\phi(J),\alpha')$ is:

$$CNA(\theta(I),\phi(J),\alpha') = \sum_{N=0}^{NMAX} (\alpha'/\alpha)^N COSGA\ F(N,\theta(I),\phi(J),\alpha)$$

where COSGA = $\cos\phi\sin\theta$, ϕ is the average azimuthal angle in the Jth azimuthal angle interval and θ is the average polar angle in the Ith polar angle interval.

In order to calculate the photon current through a plane containing both the polar and zero azimuthal axes, the equation for $CPPA(\theta(I),\phi(J),\alpha')$ is:

$$CPPA(\theta(I),\phi(J),\alpha') = \sum_{N=0}^{NMAX} (\alpha'/\alpha)^N COSGA\ F(N,\theta(I),\phi(J),\alpha)$$

where COSGA = $\sin\phi\sin\theta$, and ϕ and θ are defined as above.

The Albedo Conversion Code sums the scattered intensities and/or current for the new albedo over the polar angle groups to give the total scattered intensity or current, ASUM(ϕ(J),α') in the Jth azimuthal interval for the new albedo. In addition, the scattered intensities or current for each solid angle interval is divided by the number of steradians within the corresponding solid angle interval to put the printed scattered intensity or current for the new albedo on a per steradian basis. Thus, the printed intensity or current B(θ(I),ϕ(J),α') given by the equation

$$B(\theta(I),\phi(J),\alpha') = \frac{A(\theta(I),\phi(J),\alpha')}{STER(CTHETA(I)-CTHETA(I+1))}$$

is the scattered intensity or current per steradian at the midpoint of the Ith polar angle interval bounded by CTHETA(I)>cosθ>CTHETA(I+1) and at the midpoint of the Jth azimuthal interval. STER is the absolute value of the difference between the upper and lower bounds of the Jth azimuthal angle interval. These values, B(θ(I),ϕ(J),α'), are printed as a function of the lower bound of the polar angle interval, CTHETA(I), and the albedo α' for each azimuthal angle interval.

IV. LITE CODE UTILIZATION INSTRUCTIONS

The LITE codes are available in both ALGOL for the B-5500 and FORTRAN-IV for other computers. This section of the report includes the input data formats for the ALGOL versions of the codes. The input data formats for the FORTRAN versions are different from those in the ALGOL versions, only in that the E format in FORTRAN for floating point numbers has an E preceding the exponent, whereas the R format in ALGOL has an @ preceeding the exponent. That is, the number 217.8 would be written in R format as 2.178@+02 for the ALGOL versions of the code, and in E format as 2.178E+02 for the FORTRAN versions. The order of the input data and field width specifications is the same for both the FORTRAN and ALGOL versions.

4.1 Operator Instructions

The ALGOL versions of the LITE codes were designed to run on the Burroughs B-5500 computer. The multi-processing feature of the B-5500 allows on-line read in and printout of data from one program while computation is being performed with another program. Thus the LITE codes may be read-in and printed out on-line. The object programs may be stored on tape so that the B-5500 can read the programs from tape. Storing the object programs on tape reduces the number of cards that have to be loaded each time a program is run with one of the codes. Both the ALGOL and FORTRAN versions use one tape unit for punched output in addition to the regular input and output tape units. The punched output tape should be a BCD tape.

The running time for the LITE codes is highly dependent upon the input data. Therefore, the running time is dependent on the fraction of

the total collisions that are taken to be Rayleigh scattering events, on the average number of collisions followed per history, and on the total number of histories followed. The multi-processing feature of the B-5500 makes it difficult to predict the machine time required to run a given problem unless the problem is the only one being processed in the B-5500. The time required to run a LITE-I problem on the B-5500 was checked for three separate runs of the problem. The times required for each of the three runs were found to be different, varying by a factor of three over the range of the slowest to the fastest time. A rough estimate of the time required to run a LITE-I problem on the B-5500 may be calculated with the formula:

$$ET = 0.024(1+(ND*\overline{NPHID}))(TNC)$$

where ET is the estimated time in seconds,

ND is the number of detectors,

$\overline{NPHID}$ is the average number of azimuthal positions selected per detector for each collision, and

TNC is the total number of collisions expected for the problem.

An estimate of the time required to run a LITE-II problem on the B-5500 may be calculated with the formula:

$$ET = 0.017(1+ND)(TNC)$$

where ET, ND and TNC are defined as above.

In general the same problem run on the IBM 7090 and the Burroughs B-5500 will require 1.5 to 2 times as much time on the B-5500 as on the IBM 7090.

4.2 Input Data Formats

The input data formats for LITE-I and LITE-II are similar even though some of the input data used in LITE-I are not used in LITE-II. The input data formats will apply to both programs unless an item is followed by an asterick, and comments are made prescribing how these items should be treated when preparing input data for either LITE-I or LITE-II. The unit used to define distances (centimeters, meters, feet, etc.) should be the same for all distances described by the input data to the LITE codes. If the distance unit is meters, then the intensities are in units of photons m^{-2}/source photon for LITE-I and photons m^{-2}/unit incident flux for LITE-II. A unit incident flux is defined as one photon passing through a m^{-2} area parallel to the slab geometry.

The input for the LITE codes is divided into ten groups. The number in column 10 of the first card of each group designates the group of input data that follows on that and succeeding cards.

4.3 Control Numbers

Table I contains control numbers in Group 1 that specify the amount of input data required. Some of the control numbers appear again in the other input groups. When this occurs, the two values input for the same item must agree or the program will detect an error and terminate the problem. The number of histories to be processed, NHMAX, may be divided into sample sizes of NHMAX/NGROUP. The sample size must be less than 501. The number of groups, NGROUP, into which the histories are divided, should be large enough to provide for an accurate calculation of a standard deviation. Six bases are input for the random number generator.

This allows consecutive random numbers to be generated using a different base. Generating random numbers in this manner insures the independence between consecutive random numbers and decreases the possibility of producing identical histories when a random number generator recycles. The core storage space available limits the number of receiver positions, NDMAX, and the number of print azimuthal angle intervals, NAZA, that can be used in any one problem. The product (NDMAX*NAZA) must never be greater than 40.

TABLE I

Group 1 Input Data (Control Numbers)

Card	Format	Input Item	Definition	Limit
1	I10	LIBRAY	Input group number	=1
2	6I10	NHMAX	Number of histories	
		NGROUP	Number of deviation groups (The number of histories should be equally divisible by NGROUP.)	$\frac{NHMAX}{NGROUP} \leq 500$
		NRMAX	Number of regions	≤100
		NBMAX	Number of boundaries	≤100
		NCMAX	Maximum collisions allowed per history	
		NDMAX	Number of receivers, (NDMAX*NAZA≤40)	≤10
3	6I10	NPA	Number of print cosines	≤25
		NPCOL	Number of print collisions	≤24
		NAOP	Option for sampling source polar angles = -1, true distribution, no biasing = 0, biased sampling from isotropic distribution = 1, biasing sampling from anisotropic distribution	

TABLE 1 (continued)

Card	Format	Input Item	Definition	Limit
		NAG	Number of cosines for defining source angular distribution	≤37
		NRFLB	Number of reflection boundaries	≤5
		NMAT	Number of regions having different Mie phase functions	≤10
4	6I10	NSOREG	Number of source region	
		MAXR	Maximum number of reflections allowed	≤8
		IBASE	Base for random number generator	
		IBAS1	Base for random number generator	odd integers
		IBAS2	Base for random number generator	
		IBAS3	Base for random number generator	
5	2I10	IBAS4	Base for random number generator	
		IBAS5	Base for random number generator	

4.4 Constants

Table II contains constants in Input Group 2 that are used by the codes. Since the values to be assigned these constants depend on the individual problem, they are included as input rather than being fixed within the codes. For economy, the distance, DLONG, should be greater than the maximum possible distance within an inside region. The distance, DELTA, should be a small value, but large enough to change the maximum possible distance within an inside region in the fifth or sixth significant digit when added to that distance. ELIM is an input item that will prevent those errors that occur with a very small probability from terminating the problem. When fewer than ELIM errors occur, those errors will be listed with

TABLE II

Group 2 Input Data

Card	Format	Input Item	Definition	Limit
1	I10	LIBRAY	Input group number	=2
2	6R10.4	HS	Source height	
		DLONG	Large distance for boundary distance calculation	
		DELTA	Small distance for stepping off boundary	
		SMVAL	Small value for testing cosine and sine values to prevent division by zero	
		WCO	Weight cut-off parameter	
		ELIM	Maximum number of errors to be allowed	
3	2R10.4	DMIN	Minimum distance from collision to receiver point	
		A	Biasing parameter for sampling source azimuthal angle. (not used in LITE-II)	

the output, but only those histories containing the errors will be terminated. The results for all other histories will be saved and printed as output.

4.5 Source Angular Distribution

Input Group 3 data which are used to describe the source polar and azimuthal angle distributions are given in Table III. The source polar angle distribution is assumed to be defined with a cumulative distribution expressed in terms of the cosine of the angle measured from the positive H axis. Provisions for sampling from a biased distribution are also included to improve the sampling in the directions toward the receiver

TABLE III

Group 3 Input Data (Source Angular Distribution

Card	Format	Input Item	Definition	Limit
1	4I10	LIBRAY	Input group number	=3
		NAOP	Option for sampling source angles (See Table I)	
		NAG	Number of cosines for defining source angular distribution	
		NSAZA*	Number of angles used in describing source azimuthal angular distribution for LITE-I (leave blank for LITE-II)	
2 continues on following cards	6R10.4	CANG(J)	Cosine values at which the cumulative source polar angular probabilities are given (cosines in descending order)	J=1,NAG
Follows last card containing CANG(J)	6R10.4	PAG(J)	Cumulative probabilities defining source polar angular distribution (first value must be zero, probabilities in ascending order)	J=1,NAG
Follows last card containing PAG(J)	6R10.4	WAG(J)*	Weight parameter for biased sampling from anisotropic polar angular distribution (omit unless NAOP=1)	J=1,NAG
Follows last card containing WAG(NAG)		SAZA(J)*	Angles (degrees) used to define azimuthal angular distribution (ascending order) (omit for LITE-II)	J=1,NSAZA
		PAZA(J)*	If A≤0, cumulative azimuthal angular distribution, otherwise non-accumulative. (omit for LITE-II)	J=1,NSAZA

* WAG(J) is the weight that will be assigned to particles emitted from the source at angles with cosines between CANG(J-1) and CANG(J). Thus WAG(1) is arbitrary, since it will never be used by the code.

positions. If the original polar angle distribution is isotropic, then the program adjusts the particle weight automatically, but if the original distribution is anisotropic, then the weight adjustment parameters, WAG, must be input.

SAZA(J) and PAZA(J) are the angles and probabilities used to define the azimuthal angle distribution for LITE-I. If the value input for A in Table II is zero or negative, then PAZA(J) should be points read off the cumulative probability distribution curve. If A>0, then PAZA(J) should be points read off the non-accumulative azimuthal angular density curve.

4.6 Reflection Distribution

Table IV lists Input Group 4 data which are used in describing the reflection of light from ground and/or cloud surfaces. If the problem contains no reflection surfaces, this group of data may be omitted. A listing of Input Group 4 data is required for each reflection surface. The reflection surfaces are limited to 2 for any one problem and the boundary number assigned to any reflection surface must be less than or equal to 5. Reflection is limited to plane surfaces. The polar angle distribution of the reflected light must be expressed in terms of the cosine of the angle measured from the normal to the reflection surface and is assumed azimuthally symmetric. If the reflection distribution is isotropic in the upper or lower hemispheres, then the reflection angle distribution tables should be omitted. If the reflection distribution is anisotropic, then both the reflection distribution and the cumulative distribution must be input. The reflected distribution POR(NRB,J) is defined as the probability that a photon reflected from surface NRB will be moving with

TABLE IV.

Group 4 Input Data (Reflection Distributions

Card	Format	Input Item	Definition	Limit
1	5I10	LIBRAY	Input group number	=4
		NRB	Number of reflection boundary	≤5
		JREFLT(NRB)	Reflection Option = 1, reflection isotropic in upper hemisphere = 2, anisotropic in upper hemisphere = 3, isotropic in lower hemisphere = 4, anisotropic in lower hemisphere	
		NRFANG(NRB)	Number of points used to define reflection distribution at boundary NRB	≤37
		NRFCOS(NRB)	Number of cosines defining cumulative reflection distribution at boundary NRB	≤50
2	2R10.4	ALBEDO(NRB)	Reflection Albedo Constants	
		SIGNBT(NRB)	α = (ALBEDO - SIGNOT*cosθ)	
3 continues on following cards	6R10.4	*RFANG(NRB,J)	Cosines of angles used to define reflection distribution (descending order) (omit if JRFLT(NRB)=1 or 3 or if NRFANG(NRB)=0)	J=1, NRFANG(NRB)
Follows last card of RFANG's	6R10.4	*POR(NRB,J)	Probability of reflecting per unit solid angle into an angle whose cosine is RFANG(NRB,J) (Omit if JRFLT(NRB) =1 or 3) (omit if NRFANG(NRB)=0)	J=1, NRFANG(NRB)
Follows last card of POR's	6R10.4	RFLCOS(NRB,J)	Cosine values of reflection angle corresponding to the cumulative reflection distribution for values of J/NRFCOS(NRB). Input the values of RFLCOS in descending order. First cosine is input for probability = 1/NRFCOS(NRB). (Omit if JREFLT = 1 or 3).	J=1, NRFCOS(NRB)

* These values are not necessary in LITE-II; however, if NRFANG(NRB) ≠ 0, some arbitrary values must be input for these values, since the instructions for reading in these items are executed if NRFANG(NRB) is non zero.

a direction contained in a unit solid angle about the polar angle RFANG(J). The cumulative distribution is defined by evaluating the integral

$$\frac{J}{NRFCOS(NRB)} = 2\pi \int_{1}^{RFLCOS(NRB,J)} POR(NRB,J)d(\cos\theta)$$

for RFLCOS(NRB,J) when J = 1, 2, ..., NRFCOS(NRB). Thus the probability that a photon reflected by surface NRB will have a polar angle whose cosine lies in the interval [1, RFLCOS(NRB,J)] is J/NRFCOS(NRB) where NRFCOS(NRB) is the number of cosine values defining the cumulative reflection distribution for surface NRB.

4.7 Printout Control

Input Group 5 data, which describes the upper bounds of the print angle groups and the print collision numbers, are shown in Table V. The upper bounds of the print polar angles are given in terms of the cosine of the angles between the source-receiver axis and the direction of the scattered light at the receiver position for LITE-I and in terms of the cosine of the angle between the particle's direction and the normal to the receiver plane for LITE-II. The print collision numbers are the orders of scattering for which scattered light intensities are to be listed. The light intensity from all orders of scattering greater than the previous collision number up to and including the given collision number is listed opposite each print collision number. The azimuthal print angles are taken to be in degrees in LITE-I and are in terms of the cosine in LITE-II.

TABLE V.

Group 5 Input Data (Printout Control)

Card	Format	Input Item	Definition	Limit
1	3I10	LIBRAY	Input group number	=5
		NPCOL	Number of print collisions	≤24
		NPA	Number of print cosines (polar angle)	≤25
		NAZA	Number of print azimuthal intervals (NDMAX*NAZA≤40)	
2 continues on following cards	6I10	INCOL(J)	Print collision numbers (in ascending order)	J=1, NPCOL
Follows last card of INCOL's	6R10.4	*CIPA(J)	Print cosines polar distribution (descending order)	J=1, NPA
Follows last card of CIPA's	6R10.4	*CAZA(J)	Print azimuthal angles in degrees for LITE-I (ascending order). Cosines of the print azimuthal angle for LITE-II (descending order)	J=1, NAZA

* Intensities printed for CIPA(1) are for angle interval $0° \leq \theta \leq \cos^{-1}(CIPA(1))$ and intensities printed for CAZA(1) are for angle interval $0° \leq \phi \leq \cos^{-1}(CAZA(1))$

4.8 Receiver Locations

Input Group 6 data which describe the receiver locations are listed in Table VI. In tracing histories with LITE-I, all source particles are started in the zero azimuthal direction and the change in azimuthal position is recorded for each collision. Then, before an estimate of the intensity that scatters to each receiver is made, a source azimuthal angle is selected from the input azimuthal angular distribution and this angle is added to the

TABLE VI.

Group 6 Input Data (Receiver Locations)

Card	Format	Input Item	Definition	Limit
1	2I10	LIBRAY	Input group number	=6
		NDMAX	Number of receivers	≤10
2	3R10.4, I10,R10.4	HD(1)	Height of 1st receiver (altitudes in ascending order)	
		*RD(1)	Radius of 1st receiver	
		**AZD(1)	Azimuthal position of 1st receiver (degrees)	
		*NPHID(1)	Number of source azimuthal selections for 1st receiver	
		*DBSS(1)	Direct-beam source strength for 1st receiver	
3	3R10.4, I10,R10.4	HD(2)	Height of 2nd receiver	
		*RD(2)	Radius of 2nd receiver	
		**AZD(2)	Azimuthal position of 2nd receiver (degrees)	
		*NPHID(2)	Number of source azimuthal selections for 2nd receiver	
		DBSS(2)	Direct-beam source strength for 2nd receiver	
A card similar to 2 and 3 is required for each receiver				
Last card of group 6	3R10.4, I10,R10.4	HD(NDMAX)	Height of last receiver	
		*RD(NDMAX)	Radius of last receiver	
		**AZD(NDMAX)	Azimuthal position last receiver (degrees)	
		*NPHID (NDMAX)	Number of source azimuthal selections for last receiver	
		*DBSS (NDMAX)	Direct-beam source strength for last receiver	

* The NPHID(J) values are not used by LITE-II, and the RD(J) and DBSS(J) values should be input for LITE-II as discussed in Section 2.8.

** The azimuthal positions AZD(J) should not be included in the LITE-II input and the values of NPHID(J) and DBSS(J) should be shifted to the left 10 columns.

change in the particle's azimuthal position to give the azimuthal position of the collision. Several source azimuthal angles may be chosen for each collision, which, in effect, give several collisions at the same height and radius but at different azimuthal positions. The estimates of the intensities from the collisions located at the different azimuthal positions are then averaged to give the final estimates for those collisions at each of the receiver positions. The input item NPHID(J) specifies the number of source azimuthal angles that will be selected for the jth receiver point.

In LITE-II the source azimuthal angle is always taken to be 0°, therefore, no value need be input in LITE-II for the azimuthal position of the receivers, AZD(J), but a value may be input for the radial position of the receivers to be used in calculating the direct intensities.

In LITE-I, DBSS(J) is the light intensity per unit source strength emitted per unit solid angle in a direction toward the jth receiver position. LITE-I calculates the direct-beam intensity for the jth receiver position with the expression

$$DBI = (DBSS(J)e^{-RHOT})/T^2$$

where RHOT is the number of optical mean-free-path lengths between the source and the jth receiver position, and

T is the distance from the source point to the jth receiver position.

The equation used for direct-beam calculations in both LITE-I and LITE-II are identical, therefore, the direct-beam calculation is only applicable to plane parallel sources in LITE-II. For a plane parallel

source, the values input for RD(J) should be given by the expression

$$RD(J) = (HD(J) - HS)/\cos\theta_o$$

where HD(J) is the height of the Jth receiver plane,

HS is the height of the source, and

$\cos\theta_o$ is the cosine of the angle at which the source is incident upon the slab.

In addition, DBSS(J) should be input as the product of the number of particles emitted per unit area from the source plane times the secant of the source angle times the slant thickness square, T^2, between the source and receiver plane.

4.9 Geometry Description

Input Group 7 data listed in Table VII provide for the geometry description. An air-ground geometry is defined with region boundaries composed of horizontal planes and right circular vertical cylinders in LITE-I and by horizontal planes in LITE-II. The planes are identified as boundary type 1 and the cylinders as boundary type 2. For boundary type 1, COEE is the H intercept of the plane, and for boundary type 2, COEE is the radius of the cylindrical surface. All reflection surfaces must be assigned boundary numbers less than or equal to 5. A negative sign preceding the boundary number, NBOUND, denotes a reflection boundary. Regions are defined by the signed boundary numbers encompassing the region. In reference to planes, the minus sign denotes a "lower" plane, and the plus sign denotes an "upper" plane. In reference to a cylindrical surface, the minus sign denotes an "inner" surface, and the plus sign denotes an "outer" surface

TABLE VII.

Group 7 Input Data (Geometry Description)

Card	Format	Input Item	Definition	Limit
1	3I10	LIBRAY	Input group number	=7
		NBMAX	Number of boundaries	≤100
		NRMAX	Number of regions	≤100
2	2I10 R10.4	*NBOUND(1)	Position of boundary 1 in boundary table	
		ITYPE(1)	Type of boundary 1, ITYPE(1)=1, H plane ITYPE(1)=2, cylinder	
		COEE(1)	Coefficient of boundary 1	
	A card similar to card 2 is required for each boundary.			
Follows last boundary card	3I5, R5.2	*NREG(1)	Position of region 1 in region table	
	8I5	NB(1)	Number of boundaries encompassing region 1	
		MAT(1)	Phase function number for region 1	
		EMP(1)	Importance number for region 1	
		IB(1,1)	First boundary, bounding region 1 (sign on IB designates inner or outer boundary with respect to region 1)	
		MPR(1,1)	Most probable region of entry across first boundary of region 1	
		IB(1,2)	Second boundary bounding region 1 with appropriate sign	
		MPR(1,2)	Most probable region of entry across second boundary of region 1	
		IB(1,3)	Third boundary bounding region 1 with appropriate sign	
		MPR(1,3)	Most probable region of entry across third boundary of region 1	

TABLE VII. (continued)

Card	Format	Input Item	Definition	Limit
		IB(1,4)	Fourth boundary bounding region 1 with appropriate sign	
		MPR(1,4)	Most probable region of entry across fourth boundary of region 1	
A card similar to the preceding card is required for each region including outside regions.				

* Boundaries and regions are assigned numbers sequentially in the order they are listed in the input. The values NBOUND(J) and NREG(J) therefore should both begin with 1 for the first boundary or region listed and increase sequentially for the remaining boundaries or regions.

All space must be identified including outside regions which are not completely encompassed by boundaries. The most probable regions of entry, MPR, are given to speed up the region search process. When there are two or more possible regions of entry across a given boundary, the region with the smallest region number should be given as the most probable region of entry.

The region importance number, EMP, is given to reduce the sampling in regions of minor importance. A particle when crossing from one region to a region of more importance will not be affected by the region importance numbers. However, when a particle crosses from a given region to another region of less importance, a random number will be generated and the history terminated if the ratio of the importance numbers (EMP for region entered/EMP for region exited) is less than the random number. If the ratio of the importance numbers is greater than the random number, then the particle weight is multiplied by the reciprocal of the ratio and tracing of the history is continued.

4.10 Aerosol Scattering Data

The Input Group 8 data listed in Table VIII define the aerosol particle scattering phase functions to be used in the air-ground geometry. The data shown in Table VIII for Input Group 8 must be repeated for each phase function to be defined. Up to 10 phase functions may be defined in any one problem. MAT is the number assigned to the phase function defined by the data in Input Group 8. This number is used to designate the phase function for each of the regions defined by the Input Group 7 data.

Special routines have been incorporated into the code for treating Rayleigh scattering, therefore, it is only necessary to input the aerosol scattering phase functions. If only Rayleigh scattering is to be considered (RAYLEE = 1.0), then Input Group 8 data defining DIFCOS(MAT,J) PDCOS(MAT,J) and PHANG(MAT,J) may be omitted. The machine codes, RRA-42 and RRA-45, described in Ref. 3, can be used to compute the values to be input for the parameters PDCOS, DIFCOS and PHANG.

4.11 Cross Section Input Data

Input Group 9 data listed in Table IX give the distance in mean free paths from ground level, the ratio of the scattering-to-total cross section, and the ratio of Rayleigh-to-scattering cross section as a function of altitude. The scattering cross section is taken to be the sum of the aerosol and Rayleigh scattering cross sections. The difference between the extinction coefficient (total cross section) and the scattering cross section is defined as the absorption cross section.

TABLE VIII.

Group 8 Input Data (Aerosol Scattering Data)

Card	Format	Input Item	Definition	Limit
1	2I10	LIBRAY	Input group number	=8
		MAT	Aerosol scattering phase function number for the following data	≤10
2	2I10 10X,R10.4	NDFCOS (MAT)	Number of cosines for which the aerosol scattering phase function are given	≤50
		NPHANG (MAT)	Number of cosines used to describe the cumulative angular distributions for aerosol scattering	≤50
		RAYLEE(MAT)	= 1.0, Rayleigh scattering only = 0.0, Both Rayleigh and aerosol scattering	
3	6R10.4	*DIFCOS (MAT,J)	Cosine values at which aerosol scattering phase functions are listed. (descending order) Omit if RAYLEE = 1.0. (Omit if NDFCOS(MAT) = 0)	J=1, NDFCOS (MAT)
Follows last DIFCOS card	6R10.4	*PDCOS	Values of the phase function at the designated cosines. Omit if RAYLEE = 1.0. NDFCOS values. (Omit if NDFCOS(MAT) = 0)	J=1, NDFCOS (MAT)
Follows last PDCOS card	6R10.4	PHANG (MAT,J)	Cosines at equal probability intervals describing cumulative phase function. Omit if RAYLEE = 1.0 (descending order) PHANG(MAT,1) = 1/NPHANG(MAT)	J=1, NPHANG (MAT)

* The values input for DIFCOS(MAT,J) and PDCOS(MAT,J) are not used by LITE-II. However, these values need not be removed if one wishes to use the same Group 8 input data in LITE-II that have been made up for LITE-I.

TABLE IX.

Group 9 Input Data (Cross Section Input Data)

Card	Format	Input Item	Definition	Limit
1	2I10	LIBRAY	Input group number	=9
		NOH	Number of altitudes HV at which mean-free-path distances from ground level are to be listed	≤100
2* through NOH+1	4R10.4	HV(J)*	Altitudes for which cross section data are to be listed	J=1, NOH
		TAU(J)	Mean-free-path distances for altitude HV(J)	J=1, NOH
		SCATR(J)	Ratio of scattering-to-total cross section for altitude HV(J)	J=1, NOH
		RAYR(J)	Ratio of Rayleigh-to-scattering cross section for altitude HV(J)	J=1, NOH

* Card 2 contains the four items HV(J), TAU(J), SCATR(J), and RAYR(J) for J=1; the same four items for J=2 are on the next cards, and etc.

The distance in mean free paths, TAU(J) from the ground level to height HV(J) is defined by the equation

$$TAU(J) = \int_0^{HV(J)} \Sigma_T(h)\,dh$$

where $\Sigma_T(h)$ is the extinction coefficient as a function of the altitude h.

4.12 Data Print and Check Options

Data for Input Group 10 as given in Table X are contained on a single card. This card gives the problem number and data print and check options. The problem number is printed on output to identify the output data. IDUMP is a print option that allows the printout of intermediate values calculated during the generation of each history. This option is included to

TABLE X.

Card	Format	Input Item	Definition	Limit
1	4I10	LIBRAY	Input group number	=10
		NPROB	Problem number	
		IDUMP	Option for intermediate printout = 0, no intermediate printout = 1, gives intermediate printout	
		ICHECK	Option for checking input data = 0, no check on input data = 1, check input data	

aid in checkout. The quantity of printout produced when IDUMP is non-zero makes it inadvisable to print the intermediate data if more than ten histories are being processed.

ICHECK is an option that provides for several checks on the input data. The input cumulative probability tables are checked for ascending order, and several of the cosine tables are checked for descending order. In addition, various input values are checked to insure that storage locations reserved for dimensioned variables are not exceeded. Cards within the input data groups 1 through 9 must be arranged in the order specified in Tables I through IX, but it is not necessary to order the groups. The cards for Input Group 10 must be loaded after the cards for all other input groups have been loaded.

4.13 Loading Instructions

The LITE codes are designed to process several problems during any one computer run. The input data for a second problem may be loaded directly

behind the input data for Input Group 10 for the previous problem. Furthermore, if any of the input data groups 1 through 9 are identical for two consecutive problems, that input data group may be omitted in the second problem. Each individual problem must contain a card for Input Group 10.

V. LITE CODE SAMPLE PROBLEMS

A sample problem is included for both LITE-I and LITE-II to provide an example of the input and output formats for the two codes. The printed outputs shown in Tables XII and XIV were obtained from the FORTRAN-IV versions of the LITE codes, but the format does not vary significantly from the output format produced by the ALGOL versions. It is not possible to obtain exactly the same results when running a given problem on both the FORTRAN-IV and ALGOL versions of the LITE codes because of the differences in the random number generators used in the two versions and the differences in the word lengths for the IBM 7090 and Burroughs B-5500 computers.

5.1 LITE-I Sample Problem

The sample problem for LITE-I was designed to calculate the light intensity scattered from a point isotropic source located 807.7 meters above the ground to a point receiver placed just six meters off the ground surface at a horizontal distance of 20,120 meters away. The atmospheric model used to define the variation of the aerosol, ozone, Rayleigh and extinction coefficients with altitude is that given by Elterman (Ref. 4) for 0.65 micron wave length light. The phase function assumed for aerosol scattering was obtained from calculations reported in Ref. 3 for the "Haze C" aerosol size distribution. The ground was assumed to be a Lambert type surface reflecting light with an albedo of 0.9.

5.1.1 Input for LITE-I Sample Problem

The input for the LITE-I sample problem is given in Table XI. Bias sampling is used in sampling from both the source polar and azimuthal

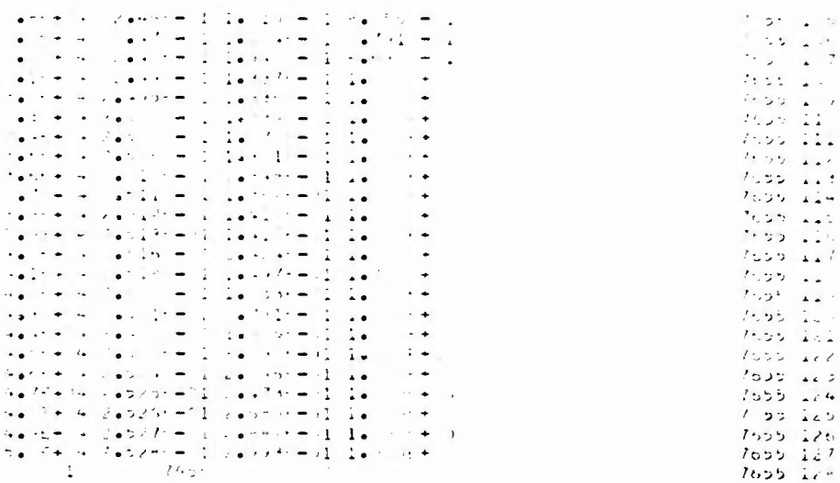
TABLE XI. LITE-I SAMPLE PROBLEM INPUT DATA

1						7[illegible]	1	
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	1			
[illegible]	[illegible]		11	1	1	7655 [illegible]		
	[illegible]	[illegible]4661	5421[illegible]47	[illegible]	762[illegible]461[illegible]	7655 [illegible]		
7[illegible]1 7	[illegible]					7655 [illegible]		
[illegible]						7655 [illegible]		
[illegible]7.7[illegible]	1.[illegible]	1.[illegible]	1.[illegible]-[illegible]	1.[illegible]-[illegible]4	[illegible]	7655 [illegible]7		
1.[illegible]	1.[illegible]					7655 [illegible]		
[illegible]		11	2			7655 [illegible]		
1.[illegible]	[illegible]	[illegible]	[illegible]	6.667E-02	0.0E+00	7655 [illegible]		
-[illegible]	-[illegible]	-[illegible]	-6.667[illegible]	-1.[illegible]		7655 011		
[illegible]	1.[illegible]	[illegible]	[illegible]	[illegible]	5.0[illegible]-01	7655 [illegible]		
[illegible]	7.[illegible]	[illegible]	[illegible]	1.[illegible]		7655 013		
[illegible]	[illegible]					7655 014		
1.[illegible]	1.[illegible]					7655 015		
[illegible]	1	2	2	2[illegible]		7655 016		
[illegible]						7655 017		
1.[illegible]	[illegible]					7655 01[illegible]		
3.[illegible]	[illegible]					7655 019		
[illegible]	[illegible]	9.[illegible]	[illegible]	[illegible]	[illegible].366E-01	7655 020		
[illegible]	[illegible].746[illegible]	7.[illegible]	7.[illegible]	6.7[illegible]	6.325E-01	7655 021		
[illegible]	[illegible]	[illegible]	4.472E-01	[illegible]	3.162E-01	7655 022		
[illegible]	[illegible]					7655 023		
[illegible]	2[illegible]	25	2			7655 024		
1	2	3	4	5	6	7655 025		
7	8	9	10	11	12	7655 026		
13	14	15	16	17	18	7655 027		
19	20					7655 028		
[illegible]	[illegible]	8.[illegible]E-01	7.00E-01	6.00E-01	5.00E-01	7655 029		
4.[illegible]E-01	3.[illegible]E-01	2.[illegible]E-01	1.[illegible]E-01	0.[illegible]E+00	-1.00E-01	7655 030		
-2.[illegible]E-01	-3.[illegible]E-01	-4.[illegible]E-01	-5.[illegible]E-01	-6.00E-01	-7.00E-01	7655 031		
-7.5[illegible]E-01	-8.[illegible]E-01	-8.5[illegible]E-01	-9.[illegible]E-01	-9.50E-01	-9.75E-01	7655 032		
-1.[illegible]E+[illegible]						7655 033		
[illegible]E+[illegible]	18[illegible]E+00					7655 034		
6	1					7655 035		
6.[illegible]E+[illegible]	2.[illegible]12E+04	0.0E+00	5	7.96E-02		7655 036		
7	2	3				7655 041		
-1	1	0.0E+00				7655 042		
2	1	5.0E+04				7655 043		
1 1	1 0.0	1 2				7655 044		
2 2	1 1.0	-1 1	2 3			7655 045		
3 1	1 0.0	-2 2				7655 046		
8	1					7655 047		
50	50	0.0E+00				7655 048		
1.000E+00	9.998E-01	9.994E-01	9.986E-01	9.976E-01	9.962E-01	76558049	1	LITE
9.945E-01	9.925E-01	9.903E-01	9.877E-01	9.816E-01	9.744E-01	76558050	2	LITE
9.659E-01	9.563E-01	9.455E-01	9.239E-01	8.870E-01	8.434E-01	76558051	3	LITE
7.660E-01	6.756E-01	5.736E-01	4.617E-01	3.420E-01	2.164E-01	76558052	4	LITE
8.716E-02	-4.362E-02	-1.736E-01	-3.007E-01	-4.226E-01	-5.373E-01	76558053	5	LITE
-6.428E-01	-7.373E-01	-8.192E-01	-8.870E-01	-9.205E-01	-9.397E-01	76558054	6	LITE

TABLE XI. (continued)

```
-9.5698E-01-9.7 3E-01-9.7 1E-01-9.7447E-01-9.8477E-01-9.9 5E-01   7655 0 5 7   LITE
-9.02 E-01-9.345E-01-9.362E-01-9.376E-01-9.386E-01-9.994E-01   7655 056 8   LITE
-9. .3E-01-1.   E+00                                           7655 057 9   LITE
 3.211E+00 3.513E+00 2.837E+00 2.193E+00 1.749E+00 1.4e5E+00   7655 0 0 1   LITE
 1.2 0E+00 1.151E+00 1.036E+00 9.45 E-01 8.167E-01 7.212E-01   7655 059 2   LITE
 6.233E-01 5.476E-01 4.940E-01 4.111E-01 3.129E-01 2.42dE-01   7655 060 3   LITE
 1.5 7E-01 1.15 E-01 8.174E-02 5.9 0E-02 4.337E-02 3.2e7E-02   7655 061 4   LITE
 .54 E-02 1.987E-02 1.692E-02 1.437E-02 1.321E-02 1.e 5E-02   7655 062 5   LITE
 .3 5E-02 1.432E-02 1.601E-02 1.982E-02 2.3 5E-02 2.497E-02   7655 063 6   LITE
 .443E-02 2.712E-02 2.814E-02 2.898E-02 2.956E-02 3.0 2E-02   7655 064 7   LITE
 3.1 6E-02 3.234E-02 3.362E-02 3.4 5E-02 3.491E-02 3.3 3E-02   7655 065 8   LITE
 3.41 E-02 3.455E-02                                           7655 066 9   LITE
 .  5E-01 9.773E-01 9. 52E-01 9.927E-01 9.  7E-01 9.  64E-01   7655 067 1   LITE
 .  4E-01 9.7 4E-01 9.745E-01 9.7 4E-01 9.650E-01 9.596E-01   7655 068 2   LITE
 9.539E-01 9.477E-01 9.412E-01 9.342E-01 9.269E-01 9.1 1E-01   7655 069 3   LITE
 9.1 3E-01 9.0 1E-01 8.924E-01 8.822E-01 8.714E-01 8.593E-01   7655 070 4   LITE
 8.478E-01 8.341E-01 8.198E-01 8.041E-01 7.873E-01 7.691E-01   7655 071 5   LITE
 7.491E-01 7.274E-01 7. 3 E-01 6.778E-01 6.4 9E-01 6.1 6E-01   7655 072 6   LITE
 5.81 E-01 5.4 8E-01 4.947E-01 4.415E-01 3.8 3E-01 3. 7 E-01   7655 073 7   LITE
 2.2 8E-01 1.126E-01-2.  7E-02-1.96 E-01-4.152E-01-6.  8E-01   7655 074 8   LITE
-3.7 6E-01-1.   E+00 3.811E+ 0 3.513E+00 2.837E+00 2.193E+00   7655 075 9   LITE
      0          51                                           7655 076
    0.0E+00    0.0E+00   1.0E+ 0 3.981E- 2                    7655 077
    1.0E+03  1.00E-01 9.974E-01 7.  3E-02                     7655 078
    2.0E+03  1.5 E-01 9.950E-01 1.521E-01                     7655 079
    3.0E+03  1.  2E-01 9.918E-01 2.774E-01                    7655 080
    4.0E+03  1.  5E-01 9.843E-01 4.354E-01                    7655 081
    5.0E+03  2.02E-01 9.768E-01 6.094E-01                     7655 082
    6.0E+03  2.07E-01 9.672E-01 8.026E-01                     7655 083
    7.0E+03  2.11E-01 9.57 E-01 9.  9E-01                     7655 084
    8.0E+03  2.14E-01 9.4 7E-01 9.623E-01                     7655 085
    9.0E+03  2.16E-01 9.280E-01 9.842E-01                     7655 086
    1.0E+04  2.19E-01 9.345E-01 9.905E-01                     7655 087
    1.1E+04  2.21E-01 8.602E-01 9.934E-01                     7655 088
    1.2E+04  2.23E-01 7.979E-01 9.894E-01                     7655 089
    1.3E+04  2.25E-01 7.137E-01 9.869E-01                     7655 090
    1.4E+04  2.27E-01 6.522E-01 9.830E-01                     7655 091
    1.5E+04  2.28E-01 6.118E-01 9.692E-01                     7655 092
    1.6E+04  2.30E-01 5.698E-01 9.431E-01                     7655 093
    1.7E+04  2.31E-01 5.194E-01 9.288E-01                     7655 094
    1.8E+04  2.33E-01 4.565E-01 9.115E-01                     7655 095
    1.9E+04  2.34E-01 3.917E-01 8.867E-01                     7655 096
    2.0E+04  2.36E-01 3.236E-01 8.750E-01                     7655 097
    2.1E+04  2.37E-01 2.  7E-01 8.622E-01                     7655 098
    2.2E+04  2.39E-01 2.  9E-01 8.453E-01                     7655 099
    2.3E+04  2.40E-01 2.055E-01 8.305E-01                     7655 100
    2.4E+04  2.42E-01 1.799E-01 8.595E-01                     7655 101
    2.5E+04  2.43E-01 1.642E-01 8.829E-01                     7655 102
    2.6E+04  2.44E-01 1.543E-01 8.991E-01                     7655 103
    2.7E+04  2.45E-01 1.533E-01 9.923E-01                     7655 104
```

TABLE XI. (continued)



angular distributions. The source azimuthal angle is chosen from the density function

$$\frac{e^{-\phi}}{1-e^{-\pi}},$$

and the polar angular distribution is input in tabular form so that the cosine of the polar angle is chosen with equal probability between the values of 1.0, 0.6667, 0.4, 0.2, 0.06667, 0,. -0.06667, -0.2, -0.4, -0.6667 and -1.0. Although only 50 histories were considered for the sample problem, the selection of five source azimuthal angles for each collision point makes the results equivalent to that which would have been obtained from running a problem for 250 histories, where only one source azimuthal angle for each collision was selected. The atmosphere is bounded by only two plane surfaces, one at the ground, h=0, and one at 50,000 meters.

5.1.2 Output for LITE-I Sample Problem

Table XII lists the output for the LITE-I sample problem. Pages 1 and 2 of Table XII give the scattered intensity as a function of collision number for the two deviation groups considered. Pages 3 and 4 of Table XII give the scattered intensities averaged over the two deviation groups and the deviation of the results for the two groups about the averaged values. Page 5 of Table XII records the number of histories terminated by each of the history termination processes and also the total number of collisions that occurred. Pages 6 and 7 of Table XII give the scattered intensity as a function of polar angle and order of reflection for the two receiver azimuthal intervals from 0° to 90° and from 90° to 180°. Page 8 of Table XII gives the scattered intensity at the receiver as a

TABLE XII. PRINTOUT FOR LITE-I SAMPLE PROBLEM

FLUXES FOR DEVIATION GROUP 1.

COLLISIONS	DETECTOR
	01
1	2.283E-11
2	6.771E-12
3	6.692E-12
4	1.442E-12
5	5.865E-13
6	3.287E-13
7	4.233E-15
8	6.318E-15
9	1.328E-14
10	2.890E-15
11	5.371E-18
12	3.522E-15
13	3.142E-16
14	3.645E-18
15	-0.
16	-0.
17	-0.
18	-0.
19	-0.
20	-0.
TOTAL	3.868E-11

BASE FOR RANDOM NUMBER GENERATOR IS5735577259

TABLE XII. (continued)

FLUXES FOR DEVIATION GROUP 2.

COLLISIONS	DETECTOR
	01
1	1.650E-11
2	9.379E-12
3	1.355E-12
4	3.059E-12
5	4.276E-13
6	2.375E-13
7	1.065E-13
8	1.723E-14
9	3.410E-15
10	1.590E-14
11	1.787E-13
12	9.026E-16
13	1.103E-14
14	7.810E-17
15	5.143E-17
16	3.837E-18
17	1.389E-17
18	8.680E-15
19	9.363E-18
20	0.
TOTAL	3.130E-11

BASE FOR RANDOM NUMBER GENERATOR IS9094901977

TABLE XII. (continued)

SCATTERED INTENSITIES VERSUS DETECTOR AND COLLISION NUMBER.

COLLISIONS	DETECTOR
	01
1	1.967E-11
2	8.075E-12
3	4.024E-12
4	2.250E-12
5	5.070E-13
6	2.831E-13
7	5.536E-14
8	1.178E-14
9	8.344E-15
10	9.396E-15
11	8.935E-14
12	2.212E-15
13	5.671E-15
14	4.087E-17
15	2.572E-17
16	1.918E-18
17	6.947E-18
18	4.340E-15
19	4.681E-18
20	0.
TOTAL	3.499E-11

BASE FOR RANDOM NUMBER GENERATOR IS9094901977

TABLE XII. (continued)

INTENSITY DEVIATIONS VERSUS DETECTOR AND COLLISION NUMBER.

COLLISIONS	DETECTOR
	01
1	2.739E-12
2	9.721E-13
3	1.887E-12
4	5.718E-13
5	5.619E-14
6	3.724E-14
7	3.615E-14
8	3.860E-15
9	3.488E-15
10	4.600E-15
11	6.317E-14
12	9.261E-16
13	3.788E-15
14	2.632E-17
15	1.818E-17
16	1.357E-18
17	4.912E-18
18	3.069E-15
19	3.310E-18
20	-0.
TOTAL	2.610E-12

BASE FOR RANDOM NUMBER GENERATOR 1S9094901977

TABLE XII. (continued)

```
RADIATION RESEARCH ASSOCIATES LITE-I PROBLEM       7655

HISTORY TERMINATION COUNTERS.

        0 HISTORIES WERE TERMINATED WHEN THE COLLISION NUMBER EXCEEDED    20.
        0 HISTORIES WERE TERMINATED BY THE REGION IMPORTANCE PARAMETERS.
       50 HISTORIES WERE TERMINATED BY MINIMUM WEIGHT CUTOFF.
        0 HISTORIES WERE TERMINATED AFTER  MAXIMUM NUMBER OF REFLECTIONS.

      397 COLLISIONS OCCURRED.
```

TABLE XII. (continued)

RADIATION RESEARCH ASSOCIATES LITE-1 PROBLEM 7655

SCATTERED LIGHT INTENSITY VERSUS ANGLE AND NUMBER OF REFLECTIONS FROM SURFACE ONE.

AZIMUTHAL RANGE = 0. TO 9.000E 01

SOURCE HEIGHT H= 8.077E 02. DETECTOR COORDINATES HD= 6.096E 00 RD=. 2.012E 04

ANGLE (COSINE)	COLLISION 0	1	2	3	4	5	TOTAL
0.9500	7.902E-12	1.676E-12	8.735E-14	3.673E-15	9.015E-18	1.358E-18	9.668E-12
0.9000	7.821E-16	2.925E-14	3.039E-14	1.661E-15	2.187E-20	2.697E-19	6.208E-14
0.8000	4.066E-15	9.783E-15	1.088E-14	5.481E-15	4.762E-18	1.619E-19	3.022E-14
0.7000	0.	3.365E-14	7.684E-17	8.001E-19	4.496E-16	2.581E-19	3.417E-14
0.6000	2.259E-17	8.417E-14	4.614E-18	1.116E-18	3.580E-19	6.853E-19	8.420E-14
0.5000	0.	8.274E-19	3.929E-20	1.299E-20	1.358E-18	5.580E-19	2.796E-18
0.4000	0.	1.177E-19	7.635E-20	2.430E-21	1.269E-18	0.	1.465E-18
0.3000	2.241E-16	1.588E-20	1.504E-17	5.822E-22	0.	1.380E-20	2.392E-16
0.2000	0.	6.867E-17	9.690E-25	4.105E-20	0.	2.550E-20	6.874E-17
0.1000	0.	1.078E-18	2.292E-20	5.894E-20	0.	0.	1.160E-18
0.	0.	7.334E-21	3.314E-19	0.	0.	0.	3.387E-19
-0.1000	0.	0.	6.204E-24	0.	0.	0.	6.204E-24
-0.2000	0.	0.	0.	0.	0.	0.	0.
-0.3000	0.	0.	0.	0.	0.	0.	0.
-0.4000	0.	0.	0.	0.	0.	0.	0.
-0.5000	0.	0.	0.	0.	0.	0.	0.
-0.6000	0.	0.	0.	0.	0.	0.	0.
-0.7000	0.	0.	0.	0.	0.	0.	0.
-0.7500	0.	0.	0.	0.	0.	0.	0.
-0.8000	0.	0.	0.	0.	0.	0.	0.
-0.8500	0.	0.	0.	0.	0.	0.	0.
-0.9000	0.	0.	0.	0.	0.	0.	0.
-0.9500	0.	0.	0.	0.	0.	0.	0.
-0.9750	0.	0.	0.	0.	0.	0.	0.
-1.0000	0.	0.	0.	0.	0.	0.	0.
TOTAL	7.907E-12	1.833E-12	1.287E-13	1.082E-14	4.664E-16	3.330E-18	9.879E-12

TABLE XII. (continued)

RADIATION RESEARCH ASSOCIATES LITE-1 PROBLEM 7655

SCATTERED LIGHT INTENSITY VERSUS ANGLE AND NUMBER OF REFLECTIONS FROM SURFACE ONE.

AZIMUTHAL RANGE = 9.000E 01 TO 1.800E 02

SOURCE HEIGHT H= 8.077E 02. DETECTOR COORDINATES HD= 6.096E 00 RD=. 2.012E 04

ANGLE (COSINE)	COLLISION 0	1	2	3	4	5	TOTAL
0.9500	1.145E-11	4.275E-12	4.341E-14	2.109E-15	1.292E-15	0.	1.577E-11
0.9000	1.060E-14	2.465E-12	6.076E-17	5.508E-16	1.608E-16	2.092E-19	2.477E-12
0.8000	1.056E-12	1.077E-12	1.009E-13	1.593E-16	3.854E-16	4.536E-20	2.234E-12
0.7000	5.193E-14	4.349E-14	6.069E-15	8.975E-14	1.589E-17	0.	1.913E-13
0.6000	9.990E-13	5.820E-14	1.101E-15	5.440E-15	1.758E-21	4.546E-19	1.064E-12
0.5000	3.093E-13	1.057E-13	3.345E-18	3.937E-21	5.494E-15	0.	4.205E-13
0.4000	4.513E-14	1.021E-13	1.601E-14	6.342E-18	2.348E-17	0.	1.632E-13
0.3000	4.828E-14	5.431E-13	8.555E-17	4.209E-17	5.148E-17	2.430E-18	5.916E-13
0.2000	1.785E-13	1.750E-14	1.180E-17	8.910E-17	4.194E-18	5.536E-20	1.961E-13
0.1000	1.028E-15	1.885E-14	4.660E-19	0.	4.247E-16	6.627E-18	2.031E-14
0.	1.353E-14	1.099E-14	4.411E-16	8.795E-18	0.	5.357E-20	2.496E-14
-0.1000	1.628E-16	1.874E-14	5.969E-16	1.999E-19	5.846E-20	0.	1.950E-14
-0.2000	6.740E-15	2.313E-13	6.893E-17	3.437E-18	6.386E-21	0.	2.381E-13
-0.3000	1.390E-14	6.919E-15	1.110E-14	5.242E-17	2.090E-20	0.	3.198E-14
-0.4000	3.745E-14	1.931E-13	1.003E-14	1.522E-19	2.811E-19	0.	2.405E-13
-0.5000	8.511E-14	8.938E-14	1.139E-18	9.262E-18	0.	1.949E-18	1.745E-13
-0.6000	1.753E-14	1.581E-13	8.298E-18	8.500E-17	2.301E-17	1.726E-19	1.757E-13
-0.7000	5.798E-14	3.867E-16	1.926E-16	9.852E-18	1.066E-16	1.619E-19	5.868E-14
-0.7500	2.356E-14	2.862E-17	4.911E-19	3.931E-19	1.006E-26	0.	2.359E-14
-0.8000	3.837E-14	9.838E-15	5.323E-19	0.	6.319E-21	0.	4.821E-14
-0.8500	2.957E-14	2.397E-13	7.224E-21	1.354E-17	0.	0.	2.692E-13
-0.9000	3.432E-15	2.486E-16	0.	1.141E-18	0.	4.339E-15	8.020E-15
-0.9500	2.647E-14	5.849E-13	1.171E-17	1.887E-17	2.797E-20	8.820E-19	6.114E-13
-0.9750	3.362E-14	1.418E-14	3.455E-15	5.519E-22	0.	1.428E-19	5.125E-14
-1.0000	5.160E-15	2.813E-15	2.904E-17	6.454E-18	0.	1.215E-19	8.009E-15
TOTAL	1.454E-11	1.027E-11	1.936E-13	9.836E-14	7.982E-15	4.352E-15	2.511E-11

TABLE XII. (continued)

RADIATION RESEARCH ASSOCIATES LITE-1 PROBLEM 7655

SCATTERED LIGHT INTENSITY VERSUS REGION OF SCATTER

REGION	DETECTOR
	01
1	0.
2	3.499E-11
3	0.
TOTAL	3.499E-11

TABLE XII. (continued)

LIGHT SCATTERED FROM REFLECTION SURFACES TO EACH DETECTOR.

NO OF REFLECTIONS	DETECTOR
	1
1	7.1765E-12
2	3.9836E-13
3	4.9150E-14
4	2.3075E-14
5	9.5953E-17
TOTAL	1.5294E-13

TABLE XII. (continued)

RADIATION RESEARCH ASSOCIATES LITE-1 PROBLEM 7655

SCATTERED LIGHT INTENSITY VERSUS ANGLE AND NUMBER OF REFLECTIONS FROM SURFACE ONE.

AZIMUTHAL RANGE = 0. TO 1.800E 02

SOURCE HEIGHT H= 8.077E 02. DETECTOR COORDINATES HD= 6.096E 00 RD=. 2.012E 04

ANGLE (COSINE)	COLLISION 0	1	2	3	4	5	TOTAL
0.9500	1.935E-11	5.950E-12	1.308E-13	5.782E-15	1.301E-15	1.358E-18	2.544E-11
0.9000	1.138E-14	2.495E-12	3.046E-14	2.212E-15	1.608E-16	4.790E-19	2.539E-12
0.8000	1.060E-12	1.086E-12	1.118E-13	5.640E-15	3.901E-16	2.073E-19	2.264E-12
0.7000	5.193E-14	7.714E-14	6.146E-15	8.975E-14	4.655E-16	2.581E-19	2.254E-13
0.6000	9.990E-13	1.474E-13	1.105E-15	5.441E-15	3.597E-19	1.140E-18	1.148E-12
0.5000	3.093E-13	1.057E-13	3.384E-18	1.692E-20	5.495E-15	5.580E-19	4.205E-13
0.4000	4.513E-14	1.021E-13	1.601E-14	6.344E-18	2.475E-17	0.	1.632E-13
0.3000	4.850E-14	5.431E-13	1.006E-16	4.209E-17	5.148E-17	2.444E-18	5.918E-13
0.2000	1.785E-13	1.757E-14	1.180E-17	8.915E-17	4.194E-18	8.086E-20	1.962E-13
0.1000	1.028E-15	1.885E-14	4.889E-19	5.894E-20	4.247E-16	6.627E-18	2.031E-14
0.	1.353E-14	1.099E-14	4.414E-16	8.795E-18	0.	5.357E-20	2.496E-14
-0.1000	1.628E-16	1.874E-14	5.969E-16	1.999E-19	5.846E-20	0.	1.950E-14
-0.2000	6.740E-15	2.313E-13	6.893E-17	3.437E-18	6.386E-21	0.	2.381E-13
-0.3000	1.390E-14	6.919E-15	1.110E-14	5.242E-17	2.090E-20	0.	3.198E-14
-0.4000	3.745E-14	1.931E-13	1.003E-14	1.522E-19	2.811E-19	0.	2.405E-13
-0.5000	8.511E-14	8.938E-14	1.139E-18	9.262E-18	0.	1.949E-18	1.745E-13
-0.6000	1.753E-14	1.581E-13	8.298E-18	8.500E-17	2.301E-17	1.726E-19	1.757E-13
-0.7000	5.798E-14	3.867E-16	1.926E-16	9.852E-18	1.066E-16	1.619E-19	5.868E-14
-0.7500	2.356E-14	2.862E-17	4.911E-19	3.931E-19	1.006E-26	0.	2.359E-14
-0.8000	3.837E-14	9.838E-15	5.323E-19	0.	6.319E-21	0.	4.821E-14
-0.8500	2.957E-14	2.397E-13	7.224E-21	1.354E-17	0.	0.	2.692E-13
-0.9000	3.432E-15	2.486E-16	0.	1.141E-18	0.	4.339E-15	8.020E-15
-0.9500	2.647E-14	5.849E-13	1.171E-17	1.887E-17	2.797E-20	8.820E-19	6.114E-13
-0.9750	3.362E-14	1.418E-14	3.455E-15	5.519E-22	0.	1.478E-19	5.125E-14
-1.0000	5.160E-15	2.813E-15	2.904E-17	6.454E-18	0.	1.215E-19	8.009E-15
TOTAL	2.245E-11	1.210E-11	3.223E-13	1.092E-13	8.448E-15	4.355E-15	3.499E-11

TABLE XII. (continued)

RADIATION RESEARCH ASSOCIATES -LITE- PROBLEM 7655

DIRECT BEAM LIGHT INTENSITIES

DETECTOR	DIRECT INTENSITY
1	2.231E-11

funtion of the region of scatter and page 9 gives the reflected intensity as a function of the order of reflection from the ground surface. Page 10 of Table XII gives the scattered intensity summed over all azimuthal angles as a function of polar angle and order of reflection. Lastly, page 11 gives the direct intensity at the receiver position.

5.2 LITE-II Sample Problem

A sample problem for the LITE-II code was designed to calculate the scattered light intensity one kilometer above the ground surface due to a plane parallel 0.5 micron wave length light source incident at 30° from the normal to the top of the atmosphere. Elterman's (Ref. 4) clear standard atmosphere model for 0.5 micron wave length light was used to define the variation of the aerosol, ozone, Rayleigh and extinction coefficients with altitude. The aerosol phase function used in the sample problem was obtained from the calculations reported in Ref. 3 for the "Haze C" aerosol size distribution. The ground was assumed to be a Lambert type surface reflecting light with an albedo of 0.9.

5.2.1 Input for LITE-II Sample Problem

Table XIII lists the input data for the LITE-II sample problem. The input is for the FORTRAN-IV version rather than for the ALGOL version. The only difference in the input for the two versions is that the ALGOL version requires that the @ symbol preceed the exponent of those numbers input with the E format. The information appearing in columns 66 through 80 of each card is not read by the program but is given as an aid in identifying the problem deck. The atmosphere is divided into laterally infinite slab regions bounded on the top by planes at 2, 4, 10, 20, 30,

TABLE XIII. LITE-II SAMPLE PROBLEM INPUT DATA

```
         1                                                                                                  5555001 LITE-II
        50           2           9           8          20           1                                     5555002 LITE-II
        25          20          -1           2           1           1                                     5555003 LITE-II
         8           6  37215983  76987433  91824651  13789347                                           5555004 LITE-II
  62138697  21964385                                                                                      5555005 LITE-II
         2                                                                                                  5555006 LITE-II
 4.999+04   1.0+06    1.0-01    1.0-04    5.0-04    5.0+01                                                5555007 LITE-II
   1.0+00                                                                                                 5555008 LITE-II
         3          -1           2                                                                        5555009 LITE-II
-8.6603-01-8.6603-01                                                                                      5555010 LITE-II
   0.0+00   1.0+00                                                                                       5555011 LITE-II
         4           1           2           0          20                                               5555012 LITE-II
   9.0-01                                                                                                 5555013 LITE-II
 9.746-01  9.437-01  9.220-01  8.944-01  8.660-01  8.366-01                                               5555014 LITE-II
 8.063-01  7.746-01  7.416-01  7.071-01  6.709-01  6.325-01                                               5555015 LITE-II
 5.915-01  5.474-01  5.000-01  4.472-01  3.873-01  3.162-01                                               5555016 LITE-II
 2.236-01  0.000+00                                                                                       5555017 LITE-II
         5          20          25           1                                                           5555018 LITE-II
         1           2           3           4           5           6                                   5555019 LITE-II
         7           8           9          10          11          12                                   5555020 LITE-II
        13          14          15          16          17          18                                   5555021 LITE-II
        19          20                                                                                    5555022 LITE-II
 9.75-01   9.50-01   9.00-01   8.50-01   8.00-01   7.50-01                                               5555023 LITE-II
 7.00-01   6.00-01   5.00-01   4.00-01   3.00-01   2.00-01                                               5555024 LITE-II
 1.00-01   0.00+00  -1.00-01  -2.00-01  -3.00-01  -4.00-01                                               5555025 LITE-II
-5.00-01  -6.00-01  -7.00-01  -8.00-01  -9.00-01  -9.50-01                                               5555026 LITE-II
-1.00+00                                                                                                  5555027 LITE-II
-1.00+00                                                                                                  5555028 LITE-II
         6           1                                                                                    5555030 LITE-II
   1.0+03  2.865+04           1 3.3333+09                                                                5555032 LITE-II
         7           8           9                                                                        5555035 LITE-II
        -1           1   0.0+00                                                                           5555036 LITE-II
         2           1   2.0+03                                                                           5555037 LITE-II
         3           1   4.0+03                                                                           5555038 LITE-II
         4           1   1.0+04                                                                           5555039 LITE-II
         5           1   2.0+04                                                                           5555040 LITE-II
         6           1   3.0+04                                                                           5555041 LITE-II
         7           1   4.0+04                                                                           5555042 LITE-II
         8           1   5.0+04                                                                           5555043 LITE-II
 1   1   1  0.0   1   2                                                                                   5555044 LITE-II
 2   2   1  1.0  -1   1   2   3                                                                           5555045 LITE-II
 3   2   1  1.0  -2   2   3   4                                                                           5555046 LITE-II
 4   2   1  1.0  -3   3   4   5                                                                           5555047 LITE-II
 5   2   1  1.0  -4   4   5   6                                                                           5555048 LITE-II
 6   2   1  1.0  -5   5   6   7                                                                           5555049 LITE-II
 7   2   1  1.0  -6   6   7   8                                                                           5555050 LITE-II
 8   2   1  1.0  -7   7   8   9                                                                           5555051 LITE-II
 9   1   1  0.0  -8   8                                                                                   5555052 LITE-II
         8           1                                                                                    5555053 LITE-II
         0          50   0.0+00                                                                          5555054 LITE-II
```

TABLE XIII. (continued)

9.991-01	9.976-01	9.956-01	9.932-01	9.904-01	9.873-01	5555055 LITE-II
9.837-01	9.799-01	9.757-01	9.713-01	9.665-01	9.613-01	5555056 LITE-II
9.557-01	9.497-01	9.434-01	9.366-01	9.295-01	9.219-01	5555057 LITE-II
9.137-01	9.051-01	8.958-01	8.861-01	8.755-01	8.644-01	5555058 LITE-II
8.524-01	8.398-01	8.261-01	8.114-01	7.957-01	7.785-01	5555059 LITE-II
7.601-01	7.401-01	7.180-01	6.939-01	6.674-01	6.380-01	5555060 LITE-II
6.051-01	5.673-01	5.255-01	4.769-01	4.208-01	3.537-01	5555061 LITE-II
2.733-01	1.746-01	4.767-02	-1.174-01	-3.381-01	-6.132-01	5555062 LITE-II
-9.69?-01	-1.?10+00					5555063 LITE-II
	52					5555064 LITE-II
-2.0+00	0.0+00	1.0+00	9.317-02			5555065 LITE-II
0.0+00	0.0+00	1.0+00	9.317-02			5555066 LITE-II
1.0+03	1.?7-01	9.995-01	1.748-01			5555067 LITE-II
2.0+03	2.34-01	9.978-01	3.039-01			5555068 LITE-II
3.0+03	2.40-01	9.?77-01	4.874-01			5555069 LITE-II
4.0+03	2.62-01	9.938-01	6.564-01			5555070 LITE-II
5.0+03	2.78-01	9.998-01	7.943-01			5555071 LITE-II
6.0+03	2.89-01	9.965-01	9.097-01			5555072 LITE-II
7.0+03	2.99-01	9.917-01	9.612-01			5555073 LITE-II
8.0+03	3.07-01	9.896-01	9.844-01			5555074 LITE-II
9.0+03	3.14-01	9.856-01	9.936-01			5555075 LITE-II
1.0+04	3.20-01	9.804-01	9.962-01			5555076 LITE-II
1.1+04	3.26-01	9.896-01	9.981-01			5555077 LITE-II
1.2+04	3.31-01	9.537-01	9.959-01			5555078 LITE-II
1.3+04	3.35-01	9.269-01	9.947-01			5555079 LITE-II
1.4+04	3.39-01	9.076-01	9.932-01			5555080 LITE-II
1.5+04	3.42-01	8.854-01	9.873-01			5555081 LITE-II
1.6+04	3.45-01	8.719-01	9.761-01			5555082 LITE-II
1.7+04	3.48-01	8.442-01	9.698-01			5555083 LITE-II
1.8+04	3.50-01	8.047-01	9.622-01			5555084 LITE-II
1.9+04	3.52-01	7.584-01	9.510-01			5555085 LITE-II
2.0+04	3.54-01	7.065-01	9.458-01			5555086 LITE-II
2.1+04	3.56-01	6.415-01	9.389-01			5555087 LITE-II
2.2+04	3.58-01	5.580-01	9.311-01			5555088 LITE-II
2.3+04	3.59-01	5.426-01	9.238-01			5555089 LITE-II
2.4+04	3.61-01	5.115-01	9.381-01			5555090 LITE-II
2.5+04	3.62-01	4.858-01	9.491-01			5555091 LITE-II
2.6+04	3.63-01	4.730-01	9.567-01			5555092 LITE-II
2.7+04	3.64-01	4.496-01	9.951-01			5555093 LITE-II
2.8+04	3.65-01	4.655-01	9.502-01			5555094 LITE-II
2.9+04	3.66-01	4.622-01	9.474-01			5555095 LITE-II
3.0+04	3.66-01	4.679-01	9.419-01			5555096 LITE-II
3.1+04	3.67-01	4.457-01	1.000+00			5555097 LITE-II
3.2+04	3.67-01	4.466-01	1.000+00			5555098 LITE-II
3.3+04	3.68-01	4.466-01	1.000+00			5555099 LITE-II
3.4+04	3.68-01	4.541-01	1.000+00			5555100 LITE-II
3.5+04	3.69-01	4.438-01	1.000+00			5555101 LITE-II
3.6+04	3.69-01	4.496-01	1.000+00			5555102 LITE-II
3.7+04	3.69-01	4.543-01	1.000+00			5555103 LITE-II
3.8+04	3.69-01	4.640-01	1.000+00			5555104 LITE-II

TABLE XIII. (continued)

[illegible]	[illegible]	[illegible]	[illegible]	5555135 LITE-11
[illegible]	[illegible]	[illegible]	[illegible]	5555136 LITE-11
[illegible]	[illegible]	[illegible]	[illegible]	5555137 LITE-11
[illegible]	[illegible]	[illegible]	[illegible]	5555106 LITE-11
[illegible]	[illegible]	[illegible]	[illegible]	5555107 LITE-11
[illegible]	[illegible]	[illegible]	[illegible]	5555110 LITE-11
[illegible]	[illegible]	[illegible]	[illegible]	5555111 LITE-11
[illegible]	[illegible]	[illegible]	[illegible]	5555112 LITE-11
[illegible]	[illegible]	[illegible]	[illegible]	5555113 LITE-11
[illegible]	[illegible]	[illegible]	[illegible]	5555114 LITE-11
[illegible]	[illegible]	[illegible]	[illegible]	5555115 LITE-11
[illegible]	[illegible]	[illegible]	[illegible]	5555116 LITE-11
[illegible]	[illegible]	[illegible]	[illegible]	5555117 LITE-11

40 and 50 kilometers. The printout of the scattered intensity versus region of scatter for each receiver position will determine those areas of the atmosphere which contribute most significantly to the scattered intensity at the various altitudes.

5.2.2. Output for LITE-II Sample Problem

The output for the LITE-II sample problem is given in Table XIV. The first two pages give the scattered light intensity at the receiver as a function of the order of collision for each of the two groups of 25 histories run. The third page gives the average of the results of the two groups and the fourth page gives the deviation of the group results about the average results. The fifth page of Table XIV lists the number of histories terminated by each of the possible history termination processes and also the total number of collisions that occurred. The sixth page of Table XIV gives the scattered intensity at the receiver as a function of the angle measured from the normal to the receiver plane and as a function of the order of reflection from the ground surface. The seventh page of Table XIV gives the scattered light intensity as a function of the region of scatter and page eight shows the amount of light reflected from the ground surface to the receiver plane for each order of reflection. Lastly, the direct intensity at the receiver is given on page nine of Table XIV.

TABLE XIV. Printout for LITE-II Sample Problem

FLUXES FOR DEVIATION GROUP 1.

COLLISIONS

DETECTOR

	01
1	1.048E 00
2	6.297E-01
3	2.132E-01
4	1.296E-01
5	6.309E-02
6	3.177E-02
7	1.759E-02
8	7.354E-03
9	3.830E-03
10	1.305E-02
11	8.730E-03
12	4.477E-03
13	6.687E-05
14	2.806E-03
15	6.726E-04
16	3.211E-04
17	2.375E-05
18	8.080E-05
19	4.319E-05
20	-0.
TOTAL	2.175E 00

BASE FOR RANDOM NUMBER GENERATOR IS4619580853

TABLE XIV. Cont.

FLUXES FOR DEVIATION GROUP 2.

COLLISIONS	DETECTOR
	01
1	1.045E 00
2	5.375E-01
3	1.418E-01
4	8.941E-02
5	1.070E-01
6	1.544E-02
7	2.097E-02
8	8.676E-03
9	5.189E-03
10	1.174E-03
11	2.738E-04
12	5.846E-04
13	2.621E-04
14	1.582E-04
15	5.513E-05
16	0.
17	0.
18	0.
19	0.
20	0.
TOTAL	1.973E 00

BASE FOR RANDOM NUMBER GENERATOR 150651243223

TABLE XIV. Cont.

SCATTERED INTENSITIES VERSUS DETECTOR AND COLLISION NUMBER.

COLLISIONS	DETECTOR
	01
1	1.046E 00
2	5.836E-01
3	1.775E-01
4	1.095E-01
5	8.505E-02
6	2.460E-02
7	1.928E-02
8	8.015E-03
9	4.509E-03
10	7.112E-03
11	4.502E-03
12	2.531E-03
13	1.645E-04
14	1.482E-03
15	4.639E-04
16	1.606E-04
17	3.687E-05
18	4.040E-05
19	2.160E-05
20	0.
TOTAL	2.074E 00

BASE FOR RANDOM NUMBER GENERATOR IS0651243223

TABLE XIV. Cont.

INTENSITY DEVIATIONS VERSUS DETECTOR AND COLLISION NUMBER.

COLLISIONS	DETECTOR
	01
1	1.112E-03
2	4.261E-02
3	2.524E-02
4	1.422E-02
5	1.553E-02
6	5.772E-03
7	1.192E-03
8	4.673E-04
9	4.805E-04
10	4.199E-03
11	2.990E-03
12	1.376E-03
13	6.904E-05
14	9.361E-04
15	2.890E-04
16	1.135E-04
17	2.607E-05
18	2.857E-05
19	1.527E-05
20	0.
TOTAL	7.119E-02

BASE FOR RANDOM NUMBER GENERATOR IS0651243223

TABLE XIV. Cont.

```
RADIATION RESEARCH ASSOCIATES 'LITE' PROBLEM      5555

HISTORY TERMINATION COUNTERS.

        1 HISTORIES WERE TERMINATED WHEN THE COLLISION NUMBER EXCEEDED    20.
        0 HISTORIES WERE TERMINATED BY THE REGION IMPORTANCE PARAMETERS.
       48 HISTORIES WERE TERMINATED BY MINIMUM WEIGHT CUTOFF.
        1 HISTORIES WERE TERMINATED AFTER  MAXIMUM NUMBER OF REFLECTIONS.

      492 COLLISIONS OCCURRED.
```

TABLE XIV. Cont.

RADIATION RESEARCH ASSOCIATES 'LITE' PROBLEM 5555

SCATTERED LIGHT INTENSITY VERSUS ANGLE AND NUMBER OF REFLECTIONS FROM SURFACE ONE.

COSINES OF AZIMUTHAL RANGE = 1.000E 00 TO -1.000E 00

SOURCE HEIGHT H= 4.999E 04. DETECTOR COORDINATES HD= 1.000E 03 RD=. 2.865E 04

ANGLE (COSINE)	COLLISION 0	1	2	3	4	5	TOTAL
0.9750	0.	2.891E-02	7.171E-05	0.	0.	0.	2.898E-02
0.9500	0.	3.249E-03	1.519E-03	7.676E-04	0.	0.	5.536E-03
0.9000	0.	6.526E-02	7.562E-03	1.753E-03	6.627E-04	0.	7.524E-02
0.8500	0.	6.090E-02	2.661E-03	1.511E-04	1.852E-04	0.	6.390E-02
0.8000	0.	8.857E-02	3.872E-04	9.284E-04	0.	5.999E-05	8.995E-02
0.7500	0.	6.160E-02	4.888E-03	3.488E-03	3.670E-04	2.039E-05	7.036E-02
0.7000	0.	1.549E-01	1.755E-02	1.164E-03	0.	4.040E-05	1.737E-01
0.6000	0.	2.310E-01	2.645E-02	3.588E-05	0.	0.	2.575E-01
0.5000	0.	1.385E-01	2.186E-02	2.127E-04	3.535E-05	0.	1.606E-01
0.4000	0.	1.458E-01	4.007E-02	5.226E-03	1.229E-03	4.626E-05	1.923E-01
0.3000	0.	2.693E-01	1.006E-02	3.673E-03	0.	0.	2.830E-01
0.2000	0.	6.622E-02	9.044E-04	5.919E-03	0.	6.183E-05	7.310E-02
0.1000	0.	9.482E-02	0.	6.747E-05	0.	0.	9.488E-02
0.	0.	3.426E-02	1.249E-02	5.420E-05	1.814E-04	0.	4.698E-02
-0.1000	0.	2.538E-02	1.002E-03	0.	0.	2.160E-05	2.641E-02
-0.2000	4.346E-02	5.594E-02	6.728E-05	4.055E-04	0.	0.	9.988E-02
-0.3000	4.215E-02	3.385E-02	2.542E-03	1.397E-03	1.250E-05	0.	7.995E-02
-0.4000	0.	1.921E-02	4.510E-03	4.669E-04	0.	2.757E-05	2.421E-02
-0.5000	1.175E-02	7.780E-03	3.366E-04	0.	0.	0.	1.986E-02
-0.6000	0.	2.190E-02	3.828E-03	0.	0.	0.	2.573E-02
-0.7000	1.194E-02	5.986E-03	1.108E-03	1.056E-05	0.	0.	1.904E-02
-0.8000	4.734E-02	2.279E-02	1.049E-04	0.	5.298E-07	0.	7.023E-02
-0.9000	4.095E-02	8.410E-03	0.	2.847E-04	6.948E-05	0.	4.971E-02
-0.9500	2.089E-02	6.766E-03	4.853E-05	7.423E-04	5.554E-05	0.	2.850E-02
-1.0000	3.519E-03	1.082E-02	1.071E-05	0.	3.687E-05	0.	1.439E-02
TOTAL	2.220E-01	1.662E 00	1.600E-01	2.675E-02	2.836E-03	2.780E-04	2.074E 00

TABLE XIV. Cont.

RADIATION RESEARCH ASSOCIATES 'LITE' PROBLEM 5555

SCATTERED LIGHT INTENSITY VERSUS REGION OF SCATTER

REGION	DETECTOR
	01
1	0.
2	1.711E 00
3	1.476E-01
4	1.882E-01
5	1.442E-02
6	8.461E-04
7	1.182E-02
8	1.989E-06
9	0.
TOTAL	2.074E 00

TABLE XIV. Cont.

LIGHT SCATTERED FROM REFLECTION SURFACES TO EACH DETECTOR.

NO OF REFLECTIONS	DETECTOR
	1
1	1.1146E 00
2	1.0717E-01
3	1.6432E-02
4	2.2155E-03
5	1.4665E-04
TOTAL	1.2405E 00

TABLE XIV. Cont.

RADIATION RESEARCH ASSOCIATES -LITE- PROBLEM 5555

DIRECT BEAM LIGHT INTENSITIES

DETECTOR	DIRECT INTENSITY
1	7.901E-01

VI. ACC CODE UTILIZATION INSTRUCTIONS

The ACC code has been written in ALGOL for the Burroughs B-5500 and FORTRAN-IV for other computers. This section includes the input data formats for the ALGOL version of the program. The input data formats for the FORTRAN-IV version are different from those in the ALGOL version, only in that the format for floating point numbers have an E preceding the exponent, rather than an @ symbol. The punched output from the LITE programs will actually compose a large portion of the input for the ACC. However, if punched output from the LITE programs is not available, the data may be punched on cards in the format given in Table XV.

6.1 ACC Input Data Format

The input data format for the ACC input is shown in Table XV. The format in Table XV is for the first problem to be loaded on the computer. If more than one problem is to be run at a time, then the first card should be omitted from all but the first problem, and the problems loaded one behind the other. The values for CTHETA(I) and F(I,J) are those given by the LITE codes and are in the proper format for input in the ACC.

TABLE XV

ACC Input Data

Card	Format	Input Item	Definition	Limit
1	I6	NPROB*	Number of problems (The scattered intensities as a function of polar angle and order of reflection for a given azimuthal interval constitutes one problem for the ACC.)	
2	I6	IPROB	Problem number assigned to identify printed output	

TABLE XV. (continued)

Card	Format	Input Item	Definition	Limit
3	I6	NCUR	Option for determining reference plane for light current calculations. NCUR=1 Intensity but no current given NCUR=2 Intensity plus current given for plane normal to polar axis NCUR=3 Intensity plus current given for plane normal to 0 azimuthal axis NCUR=4 Intensity plus current given for plane parallel to polar and zero azimuthal axes.	
4	3I6	NANGLS	Number of cosine bounds bounding the polar angle intervals for which intensities from the LITE codes are recorded.	
		NRFLT	Number of reflection orders for which the LITE code gives the scattered light intensity. (This number includes the zeroth reflection order.)	
		NNALB	Number of new albedo values for which output is desired.	
5	3R10.4	HSORS	Source height	
		HD	Detector height	
		RD	Radial position of detector	
6	7R8.4	ALB(K)	New albedo values for which intensity or current is to be defined	K=1, NNALB
Follows last ALB(K) card	3R8.4	OALB	Albedo value at which output from LITE code is defined (old albedo)	
		ABC	If NCUR=1, ABC arbitrary If NCUR=2, ABC arbitrary If NCUR=3, ABC is the absolute value of the cosine of the midpoint of the azimuthal angle interval for which the LITE intensity is given. If NCUR=4, ABC is the sine of the midpoint of the azimuthal angle interval for which the LITE intensity is given.	

TABLE XV. (continued)

Card	Format	Input Item	Definition	Limit
		STER	Width of azimuthal interval for which intensities are given (radians)	
Follows OALB card	7R8.4	CTHETA(I)	Cosines of the upper bounds of the polar angle intervals used to define the polar angular distribution of the LITE code intensities (descending order)	I=1, NANGLS
Follows last CTHETA(I) card	6E11.4	F(I,J)	LITE code intensities given as a function of polar angle and order of reflection. I varies most rapidly. (First value for each polar angle interval begins on a new card.)	I=1, NREFLT J=1, NANGLS

* If more than one problem is to be run, the card containing NPROB should be omitted in all but the first.

VII. ACC SAMPLE PROBLEM

The sample problem for ACC is designed to convert the scattered light intensity calculated with the LITE-I code for a 0.9 ground albedo to data for albedos of 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9 and 1.0. In addition to calculating the scattered intensities for the new albedos, the scattered current across a plane normal to the polar print axis is also to be calculated. The polar axis is the line joining the source and receiver point and since the source and receiver are at the same altitude, the polar axis is parallel with the ground surface.

7.1 Input for ACC Sample Problem

The input for the ACC sample problem is shown in Table XVI. The problem is the first of 106 problems that are to be run as a group and the number of this first problem is 40107. The first eight cards in Table XVI were keypunched from information supplied on keypunch data sheets. The remainder of the cards are the punched output from a problem run with LITE-I. Only those angular intensities from 0° to 90° are used as input for the ACC sample problem; the cosines -0.1000 and -0.2000 are not read into memory. Note also that the cosines listed are the polar angle bounds so that the 12 values of the cosine from 1.0000 to 0.0000 are the bounds for the 11 polar angle intervals for which intensities are given.

7.2 Output for ACC Sample Problem

Table XVII shows the output for the ACC sample problem. The first two pages give the polar angular distribution of the scattered intensity

TABLE XVI. ACC SAMPLE PROBLEM INPUT DATA

```
  156
4.1167
    4
   12       9      10
1.0 .0+01  1.0 .0+01  1.0 .0+01
     0.1        0.2        0.3        0.4        0.5        0.6        0.7
     0.8        0.9        1.0
     0.9   1.0        6.2832
1.0000     0.9500     0.9025     0.8574     0.8145     0.7737     0.7350
0.4305     0.3000     0.2100     0.1470     0.1029     -0.1000    -0.2000
2.5272-05  5.6878-07  1.2795-07  2.2469-08  6.4597-09  7.7008-10  401 7  01 ACC
3.4597-11  5.5949-12  2.3540-13  0.0000-00                        401 7  02 ACC
9.2577-06  2.2954-06  7.6355-07  2.2769-07  6.6377-08  1.8176-08  401 7  03 ACC
1.2727-11  2.3551-13  5.1187-12  0.0000-00                        401 7  04 ACC
1.2898-05  1.1870-05  1.1508-05  1.0957-05  1.0507-05  8.5540-10  401 7  05 ACC
4.6903-11  1.2075-11  4.4138-13  0.0000-00                        401 7  06 ACC
9.2837-06  1.8760-05  1.9276-06  2.0552-07  2.0782-08  3.7567-10  401 7  07 ACC
2.6535-11  1.9490-12  0.0000-00                                   401 7  08 ACC
7.6273-06  2.0862-05  2.1710-06  3.0487-07  5.1551-08  2.2006-09  401 7  09 ACC
6.5563-11  7.9068-11  2.2216-11  0.0000-00                        401 7  10 ACC
9.2545-06  1.8695-05  1.5971-06  3.4883-07  4.4682-08  8.7542-10  401 7  11 ACC
1.9683-10  1.5773-12  2.3050-11  0.0000-00                        401 7  12 ACC
4.9259-06  1.8598-05  1.7157-06  3.4520-07  5.2851-08  5.6352-10  401 7  13 ACC
1.2436-10  9.7815-13  0.0000-00  0.0000-00                        401 7  14 ACC
3.3324-06  1.6887-05  2.2326-06  2.9192-07  3.2572-08  1.3358-09  401 7  15 ACC
1.5451-10  1.0864-13  4.6890-11  0.0000-00                        401 7  16 ACC
3.4313-06  1.1926-05  1.4098-06  3.6725-07  5.0040-08  2.4942-09  401 7  17 ACC
4.1445-11  6.2279-12  1.4051-13  0.0000-00                        401 7  18 ACC
2.9505-06  1.2987-05  2.3007-06  2.6201-07  3.5550-08  2.4389-09  401 7  19 ACC
1.1071-09  8.7153-10  2.7622-13  0.0000-00                        401 7  20 ACC
2.4992-06  1.3076-05  2.2510-06  2.9833-07  1.4604-08  2.0194-09  401 7  21 ACC
1.0233-09  1.1209-12  5.9298-11  0.0000-00                        401 7  22 ACC
```

TABLE XVII. ACC Sample Problem Output Data

RADIATION RESEARCH ASSOCIATES *ACC* PROBLEM 40107

SCATTERED LIGHT INTENSITY VERSUS ANGLE AND ALBEDO

SOURCE HEIGHT = 1.000E 01 DETECTOR COORDINATES HD= 1.000E 01 RD= 1.000E 01

ANGLE (COSINE)	ALBEDO 0.1000	0.2000	0.3000	0.4000	0.5000	0.6000	0.7000
0.9500	8.0758E-05	8.1088E-05	8.1435E-05	8.1799E-05	8.2181E-05	8.2582E-05	8.3004E-05
0.9000	3.0247E-05	3.1152E-05	3.2119E-05	3.3152E-05	3.4251E-05	3.5417E-05	3.6654E-05
0.8000	2.2142E-05	2.4136E-05	2.6183E-05	2.8286E-05	3.0449E-05	3.2677E-05	3.4974E-05
0.7000	1.8047E-05	2.1482E-05	2.5000E-05	2.8604E-05	3.2299E-05	3.6089E-05	3.9978E-05
0.6000	1.5872E-05	1.9692E-05	2.3601E-05	2.7604E-05	3.1706E-05	3.5912E-05	4.0227E-05
0.5000	1.8066E-05	2.1470E-05	2.4946E-05	2.8500E-05	3.2138E-05	3.5864E-05	3.9685E-05
0.4000	1.0983E-05	1.4363E-05	1.7824E-05	2.1363E-05	2.4990E-05	2.8711E-05	3.2533E-05
0.3000	8.3339E-06	1.1455E-05	1.4670E-05	1.7984E-05	2.1402E-05	2.4928E-05	2.8567E-05
0.2000	7.5981E-06	9.7960E-06	1.2059E-05	1.4394E-05	1.6806E-05	1.9301E-05	2.1886E-05
0.1000	7.0382E-06	9.4746E-06	1.2009E-05	1.4644E-05	1.7386E-05	2.0238E-05	2.3206E-05
0.	6.3346E-06	8.7829E-06	1.1325E-05	1.3965E-05	1.6704E-05	1.9548E-05	2.2497E-05
TOTAL	1.0677E-04	1.2364E-04	1.4099E-04	1.5885E-04	1.7725E-04	1.9620E-04	2.1575E-04

TABLE XVII. Cont.

RADIATION RESEARCH ASSOCIATES *ACC* PROBLEM 40107

SCATTERED LIGHT INTENSITY VERSUS ANGLE AND ALBEDO

SOURCE HEIGHT = 1.000E 01 DETECTOR COORDINATES HD= 1.000E 01 RD= 1.000E 01

ANGLE (COSINE)	ALBEDO 0.8000	0.9000	1.0000
0.9500	8.3446E-05	8.3911E-05	8.4400E-05
0.9000	3.7963E-05	3.9345E-05	4.0803E-05
0.8000	3.7346E-05	3.9797E-05	4.2332E-05
0.7000	4.3971E-05	4.8073E-05	5.2289E-05
0.6000	4.4657E-05	4.9206E-05	5.3832E-05
0.5000	4.3608E-05	4.7638E-05	5.1784E-05
0.4000	3.6461E-05	4.0502E-05	4.4665E-05
0.3000	3.2324E-05	3.6205E-05	4.0216E-05
0.2000	2.4568E-05	2.7353E-05	3.0250E-05
0.1000	2.6294E-05	2.9508E-05	3.2854E-05
0.	2.5558E-05	2.8732E-05	3.2024E-05
TOTAL	2.3593E-04	2.5676E-04	2.7828E-04

TABLE XVII. Cont.

RADIATION RESEARCH ASSOCIATES *ACC* PROBLEM 40107

SCATTERED LIGHT CURRENT (HOR. PLANE) VERSUS ANGLE AND ALBEDO

SOURCE HEIGHT = 1.000E 01 DETECTOR COORDINATES HD= 1.000E 01 RD= 1.000E 01

ANGLE (COSINE)	ALBEDO 0.1000	0.2000	0.3000	0.4000	0.5000	0.6000	0.7000
0.9500	7.8739E-05	7.9061E-05	7.9399E-05	7.9754E-05	8.0127E-05	8.0518E-05	8.0928E-05
0.9000	2.7979E-05	2.8815E-05	2.9710E-05	3.0665E-05	3.1682E-05	3.2761E-05	3.3905E-05
0.8000	1.8821E-05	2.0516E-05	2.2255E-05	2.4043E-05	2.5882E-05	2.7776E-05	2.9728E-05
0.7000	1.3535E-05	1.6112E-05	1.8750E-05	2.1453E-05	2.4224E-05	2.7067E-05	2.9984E-05
0.6000	1.0317E-05	1.2800E-05	1.5340E-05	1.7943E-05	2.0609E-05	2.3343E-05	2.6148E-05
0.5000	9.9360E-06	1.1804E-05	1.3720E-05	1.5675E-05	1.7676E-05	1.9725E-05	2.1827E-05
0.4000	4.9446E-06	6.4654E-06	8.0209E-06	9.6135E-06	1.1246E-05	1.2920E-05	1.4640E-05
0.3000	2.9169E-06	4.0091E-06	5.1344E-06	6.2944E-06	7.4906E-06	8.7247E-06	9.9983E-06
0.2000	1.8995E-06	2.4490E-06	3.0149E-06	3.5985E-06	4.2015E-06	4.8253E-06	5.4715E-06
0.1000	1.0557E-06	1.4213E-06	1.8013E-06	2.1967E-06	2.6079E-06	3.0358E-06	3.4809E-06
0.	3.1673E-07	4.3914E-07	5.6626E-07	6.9824E-07	8.3522E-07	9.7738E-07	1.1249E-06
TOTAL	7.3577E-05	8.1655E-05	8.9950E-05	9.8473E-05	1.0724E-04	1.1626E-04	1.2555E-04

TABLE XVII. Cont.

RADIATION RESEARCH ASSOCIATES *ACC* PROBLEM 40107

SCATTERED LIGHT CURRENT (HOR. PLANE) VERSUS ANGLE AND ALBEDO

SOURCE HEIGHT = 1.000E 01 DETECTOR COORDINATES RD= 1.000E 01 RD= 1.000E 01

ANGLE (COSINE)	ALBEDO 0.8000	0.9000	1.0000
0.9500	8.1360E-05	8.1813E-05	8.2290E-05
0.9000	3.5115E-05	3.6394E-05	3.7743E-05
0.8000	3.1744E-05	3.3827E-05	3.5982E-05
0.7000	3.2978E-05	3.6055E-05	3.9217E-05
0.6000	2.9027E-05	3.1984E-05	3.5023E-05
0.5000	2.3984E-05	2.6201E-05	2.8481E-05
0.4000	1.6407E-05	1.8226E-05	2.0099E-05
0.3000	1.1313E-05	1.2672E-05	1.4076E-05
0.2000	6.1419E-06	6.8383E-06	7.5626E-06
0.1000	3.9440E-06	4.4261E-06	4.9230E-06
0.	1.2779E-06	1.4366E-06	1.6012E-06
TOTAL	1.3512E-04	1.4500E-04	1.5519E-04

for each of the ground albedos. These intensities are the intensity per unit solid angle. On the last two pages the polar angular distribution of the current across a plane normal to the polar axis is given for each of the albedos.

VIII. PROGRAM DESCRIPTIONS

Both of the LITE codes are divided into several subroutines which are designated as procedures in the ALGOL language. The ACC is composed of a single procedure. The ALGOL programs are compiled each time they are loaded on the computer and no object decks are produced. The ALGOL language requires that any procedure called by another procedure be loaded before the calling procedure. For this reason the procedures used in LITE-I and LITE-II are listed in the following sections in reverse order with respect to the order they are executed at run time. LITE-I and LITE-II are each composed of a set of procedures that have the same names. Although procedures with the same name in the two codes are similar and perform the same function, they may not be interchangeable. The following is a listing of the procedures used in the LITE codes and a one-sentence description of each procedure.

Procedures Used in the LITE-I and LITE-II codes

Procedure	Purpose
MAIN	Reads in the input data
SRMAIN	Controls the flow of the problem on the machine
SRCHECK	Checks input data
SRDBEAM	Calculates direct intensities
SRSCTANG	Calculates scattering and direction after collision
SRREFLCT	Calculates new direction after a reflection
SRINITIAL	Initializes parameters used in accumulating the scattered intensities
SRPATHL	Generates random path lengths between collisions

Procedure	Purpose
SRANGLE	Selects source angles from input distribution
SRAVRAGE	Calculates and prints average scattered intensities as a function of collision number and receiver position over each deviation group
SRANSWER	Calculates and prints the average scattered intensities as a function of receiver position, receiver angle, and order of reflection over all histories
SRDETECT	Calculates scattered intensities at receiver points from each collision point
SRDIFSCA*	Calculates the probability of a photon scattering into a direction so as to be headed toward the receiver from each collision point and reflection surface
SRDSTBD	Calculates the distance along particles direction to boundary of region containing collision
SRSEARCH	Locates region containing the particles position coordinates for each collision
SRRANDA	Generates random numbers used in the sampling processes

* This procedure is used only in LITE-I.

8.1 ALGOL Listings for LITE-I

The following is the ALGOL listing of LITE-I. Cards 00000050 through 00043000 were furnished by the computing center at Fort Monmouth. Their purpose is to define the input-output files and to provide some of the basic functions such as tangent, exponential and etc.

```
 BEGIN FILE OUT PRINT 6 (2,15)}INTEGER XRAZQ,YVUNU,FZUVC,LKNJA,UKVUK,QNA          00000050    0000
                                                                    START OF SEGMENT ********** 0002
NI,LJLOU,GCPUV}INTEGER ARRAY ZIKLA,QNCCL (0 :12)}FORMAT MHFNK ("TIME ON           00000080    0005
                                                                    START OF SEGMENT ********** 0003
 ",I4,X4,I2,X1,A3,"  I9",A2),CMGUK I"TIME OFF ",I4,X30,"PROC. TIME =",I1          00000070    0007
0," SECS",X20,"I/O TIME =",I10," SECS")}DEFINE BLZAT =( JLOU +FZUVC DIV 2         00000080    0007
                                                                    0003 IS 002H LONG, NEXT SEG 0002
16000}GCPUV +FZUVC MOD 216000 /3600 #}FIEL ZIKLA [+]WITH 0,3I,5V,90,I2),          00000090    0007
                                                                    START OF SEGMENT ********** 0004
151,181,212,243,273,304 ,334,360}FILL QNCCL (+IWITH 0,"JAN","FEB","MAR",          00000100    0004
                                                                    0004 IS 0013 LONG, NEXT SEG 0002
                                                                    START OF SEGMENT ********** 0005
"APR","MAY","JUN","JUL","AUG","SEP","OCT","NOV","DEC"}FZUVC +TIME (1)}LK          00000110    0010
                                                                    0005 IS 0013 LONG, NEXT SEG 0002
NJA +TIME (2)}UKVUK +TIME (3)}YVUNU +TIME (0)}IF (10=YVUNU,[1816]+YVUNU,          00000120    0012
(2416])MOD 4 =0 THEN FOR XRAZQ =2 STEP 1 UNTIL 12 DO ZIKLA[XRAZQ]=ZIKLA[          00000130    0017
XRAZQ]+I }QNANI +100 XVVUNU ,[30 [4]+10 XVVUNU ,[36 [6]+VVUNU ,[42 [4]}X          00000140    0021
RAZQ +1}WHILE QNANI >ZIKLA [XRAZQ]DO XRAZQ +XRAZQ +1}QNANI +QNANI -ZIKLA          00000150    002V
 [XRAZQ =1)}PLZATIWRITE [PRINTIPAGE],MHFRK,100=LJLOU,GCPUV,QNANI,Q=CCLI=          00000160    0033
RAZQ],VVUNU,11811211)}                                                           00000170    0049
BEGIN                                                                            00001000    0055
FILE    CARD  12,10)}                                                            00002000    0055
                                                                    START OF SEGMENT ********** 0006
FILE IN CARDS (2,10)}                                                            00003000    0005
ARRAY CCLNQ[0:9]}                                                                00003010    0010
LABEL L1Z, L2Z }                                                                 00003020    0012
L1Z:  READ   (CARDS, 10, CCLNQ[+I)[L2Z] }                                        00003030    0012
      WRITE  (CARD , 10, CCLNQ[+])}                                              00003040    0016
      GO TO L1Z }                                                                00003050    0020
L2Z:  REWIND (CARD) }     CLOSE (CARDS, RELEASE) }                               00003060    0022
BEGIN                                                                            00003500    0025
SAVE FILE OUT PUNCH (2, 10, SAVE 20) }                                           00004000    0025
```

```
                                                                                START OF SEGMENT ********** 0007
FILE XXXXXX 2(2,15)$                                                                    00005000    0005
FILE TAPE1 2(2,15)$                                                                     00006000    0010
FILE TAPE2 2(2,15)$                                                                     00007000    0015
FILE TAPE3 2(2,15)$                                                                     00008000    0020
FILE TAPE4 2(2,15)$                                                                     00009000    0025
FILE TAPE5 2(2,15)$                                                                     00010000    0030
FILE TAPE6 2(2,15)$                                                                     00011000    0035
FILE TAPE7 2(2,15)$                                                                     00012000    0040
FILE TAPE8 2(2,15)$                                                                     00013000    0045
FILE TAPE9 2(2,15)$                                                                     00014000    0050
FILE TAPE10 2(2,15)$                                                                    00015000    0055
FILE TAPE11 2(2,15)$                                                                    00016000    0060
FILE TAPE12 2(2,15)$                                                                    00017000    0065
FILE TAPE13 2(2,15)$                                                                    00018000    0070
FILE TAPE14 2(2,15)$                                                                    00019000    0075
FILE TAPE15 2(2,15)$                                                                    00020000    0080
FILE TAPE16 2(2,15)$                                                                    00021000    0085
SWITCH FILE FILESW=XXXXXX,TAPE1,TAPE2,TAPE3,TAPE4,TAPE5,TAPE6,TAPE7,                    00022000    0090
   TAPE8,TAPE9,TAPE10,TAPE11,TAPE12,TAPE13,TAPE14,TAPE15,TAPE16$                        00023000    0102
LABEL FINIS$                                                                            0002A000    0113
REAL ARRAY DATA[0:63,0:511]$    COMMENT USED WITH DATA STATEMENTS ONLY$                 00025000    0113
REAL Q,XPR$    INTEGER K$                                                               00026000    0115
FORMAT F(///////"STOP / PAUSE NO. ",I5), OCTL(256D)$                                   00027000    0115
                                                                                START OF SEGMENT ********** 000F
                                                                                0008 IS 0017 LONG, NEXT SEG 0007
REAL PROCEDURE INT(ARG1)$          VALUE ARG1$          REAL ARG1$                      00028000    0115
   INT=SIGN(ARG1)*ENTIER(ABS(ARG1))$                                                    00029000    0115
REAL PROCEDURE TANH(ARG1)$         VALUE ARG1$          REAL ARG1$                      00030000    0123
   TANH=((Q=EXP(ARG1*2))-1)/(Q+1)$                                                      00031000    0123
REAL PROCEDURE MAX(ARG1,ARG2)$     VALUE ARG1,ARG2$     REAL ARG1,ARG2$                 00032000    0130
   MAX=IF ARG1≥ARG2 THEN ARG1 ELSE ARG2$                                                00033000    0130
```

```
REAL PROCEDURE MIN(ARG1,ARG2);    VALUE ARG1,ARG2;   REAL ARG1,ARG2;          00034000   0134
   MIN:=IF ARG1<ARG2 THEN ARG1 ELSE ARG2;                                     00035000   0135
REAL PROCEDURE DIM(ARG1,ARG2);    VALUE ARG1,ARG2;   REAL ARG1,ARG2;          00036000   0140
   DIM:=MAX(ARG1-ARG2,0);                                                     00037000   0140
REAL PROCEDURE TSIGN(ARG1,ARG2);  VALUE ARG1,ARG2;   REAL ARG1,ARG2;          00038000   0144
   TSIGN:=SIGN(ARG2)*ABS(ARG1);                                               00039000   0144
REAL PROCEDURE LOG(ARG1);         VALUE ARG1;        REAL ARG1;              00040000   0149
   LOG:=LN(ARG1)/2.30258509299;                                               000A1000   0149
PROCEDURE ERROR(ARG1);            VALUE ARG1;        REAL ARG1;              000A2000   0155
BEGIN WRITE(PRINT,F,ARG1); GO TO FINIS END;                                   000A3000   0155
REAL ARRAY                                                                      19000   0165
     ARC[0:201],                                                              20000   0165
     SVIFLUX[0:10,     0:40],                                                 21000   0168
      SVFLUX[0:25,0:10,0:40],                                                 22000   0170
      SVOIFCOS[0:50,0:10 ],                                                   23000   0173
      SVPDCOS [0:50,0:10 ],                                                   24000   0175
      SVPHANG [0:50,0:10 ],                                                   25000   0177
      SVAFLUX [0:25,0:10 ],                                                   26000   0179
      SVPOR   [0:37,0:5  ],                                                   27000   0182
      SVRFANG [0:37,0:5  ],                                                   28000   018A
      SVSAFLUX[0:25,0: 10],                                                   29000   0186
      SVSOFLUX[0:25,0:10 ],                                                   30000   0188
      SVFLUD  [0:100,0:10 ],                                                          0191
      SVRFLCOS[0:50,0:10 ],                                                   32000   0193
      SVA     [0:10 ],                                                        33000   0195
      SVCANG  [0:37 ],                                                        34000   0197
      SVEMP   [0:100],                                                                0199
      SVFLUR  [0:10 ],                                                        36000   0201
      SVCIPA  [0:30 ],                                                                0203
      SVFFLUX [0:10 ],                                                        38000   0205
      SVALREDO[0:5  ],                                                        39000   0207
      SVCOFE  [0:100],                                                                0209
```

```
      SVOVFLUX[0:10 ],                                    41000   0211
      SVHO    [0:10 ],                                    42000   0213
      SVPAG   [0:37 ],                                    43000   0215
      SVRAYLEE[0:10 ],                                    44000   0217
      SVSANG  [0:500],                                    45000   0219
      SVSTFLUX[0:10 ],                                    46000   0221
      SVWEIGHT[0:500],                                    47000   0223
      SVDRFLUX[0:10 ],                                    48000   0225
      SVPFANG [0:50 ],                                    49000   0227
      SVWAG   [0:37 ],                                    50000   0229
      SVPHFLT [0:50 ],                                    51000   0231
      S       [0:10 ],                                    52000   0233
      SVRFLUX [0:10 ],                                    53000   0235
      SVRO[0:25,0:10],                                     0531   0237
      SVSIGNOT[0:10 ],                                    54000   0239
      SVSUMHHO[0:50 ],                                    55000   0241
      SVCRATIO [0:10] ,                                   56000   0243
      SVHV[0:100],                                        57000   0245
      SVTAU[0:100],                                       58000   0247
SVSCATR[0:100],                                           59000   0249
      SVHAYH[0:100],                                      60000   0251
      SVTAUHO[0:10],                                      61000   0253
          SVHOO[0:10,0:10] ,                                      0255
          SVCAZA[0:50] ,                                          0258
          SVAZO[0:10] ,                                           0260
          SVSAZA[0:37] ,                                          0262
          SVPAZA[0:37] ,                                          0264
          SVCCAZA[0:50] ,                                         0266
          SVANG[0:31] ,                                           0268
      SVORSS  [0:10 ]]                                    62000   0270
INTEGER ARRAY                                             63000   0271
      SVIR    [0:4 ,0:100],                                       0271
```

```
      SVWPH   [0:4 ,0:100],                                          0274
          SVNRFB[0:100],                                             0276
          SVIINLE[0:50],                                             0278
      SVJREFLT[0:5  ],                                    66000      0280
      SVNOFCOS[0:10 ],                                    67000      0282
      SVNREG  [0:100],                                               0284
      SVINCOL [0:25 ],                                    69000      0286
      SVMAT   [0:100],                                               0288
      SVNB    [0:100],                                               0290
      SVNPHANG[0:10 ],                                    72000      0292
      SVNRFANG[0:5  ],                                    73000      0294
      SVNRICO [0:100],                                               0296
      SVITYPE [0:100],                                               0298
      SVMATERL[0:10 ],                                    76000      0300
      SVNROUND[0:100],                                               0302
      SVNPHID [0:10 ],                                    78000      0304
    NRFB[0:5],                                            79000      0306
          SVNDET[0:10] ,                                             0308
          SVIINLF[0:50] ,                                            0310
      SVNRFCOS[0:5  ]}                                    80000      0312
REAL                                                      81000      0314
    JALPHA ,   JBETA ,   JBRAC ,   JCDEPHI,   JCOTH ,    82000      0314
    JCOTHI ,   JCOTH2 ,  JCPA  ,   JCPHI  ,   JCPHII ,   83000      0314
    JCPHI2 ,   JCPHIO ,  JCPRRO ,  JCPI   ,   JCSA  ,    84000      0314
    JCSANG ,   JCTEP  ,  JDELTA ,  JDEOM  ,   JDIFM  ,   85000      0314
    JDIST  ,   JDLONG ,  JDUM   ,  JDT    ,   JEAM   ,   86000      0314
    JELIM  ,   JFI    ,  JFNPA  ,  JFNRA  ,   JH     ,   87000      0314
    JH1    ,   JH2    ,  JHS    ,  JHT    ,              88000      0314
                                    JPAG   ,   JPJMI  ,   89000      0314
    JPL    ,   JPSCAT ,  JR     ,  JR1    ,   JR2    ,   90000      0314
    JREFL  ,   JRESULT,  JRHO   ,  JRHOT  ,   JRN    ,   91000      0314
    JRRD2  ,   JRRDSQ ,  JRT    ,  JSDEPHI,   JSITH  ,   92000      0314
```

```
      JSITH1 ,    JSITH2 ,    JSMVAL ,    JS00  ,    JSPHI  ,                93000   031A
      JSPHI1 ,    JSPHI2 ,    JSPHI0 ,    JSPT  ,    JSSANG ,                94000   031A
      JSTEP  ,    JSUMDST,    JSUMSQ ,    JT    ,    JTEMP  ,                95000   031A
      JTS    ,    JUPLMIT,    JWAIT  ,    JWCO  ,    JWHUA  ,                96000   031A
    JWATLLE, JTAUM, JTAUM1, JTAUM2,                                          97000   031A
           JC0AZ1 , JS0AZI , JCAPHI , JSAPHI ,                                       031A
           JAZMAX , JC00 , JSI0 , JSAM , JRAT , JANG ,                               031A
           JCWAIT , JPAZ , JUIFANG , JCARK , JSPA , JCAP ,                           031A
           JARG , JAPA , JCAZAO , JADJUST , JPMI ,                                   031A
           JCAPHI1 , JSAPHI1 , JSRATIO ,  JSAP ,                                     031A
      Jx     ,    JXR    ,    JERRORS,    JDMIN  )                          98000   031A
  INTEGER                                                                    99000   031A
         JCMB, JJMT, JNREFL, JMAXN, JNMAXR, JIBAS1, JIBAS2,                  100000   031A
         JIBAS3, JIBASA, JIBASS, JNOM,                                       101000   031A
                                                    JIBASE ,                 102000   031A
      JICH   ,    JIDUMP ,    JJI    ,    JKA1   ,    JXA2   ,             103000   031A
      JKA3   ,    JKA4   ,    JLA    ,    JLR    ,    JLIBRAY,             104000   031A
      JLOC   ,    JLP    ,    JLSR   ,    JLST   ,    JMATI  ,             105000   031A
           JNCAND , JJJ , JJO ,                                                      031A
         JLA7 , JJADO , JJAOMAX , JKOUNT , JMMC2 ,                                   031A
      JMAT2  ,    JNAXCOL,    JMPREG ,    JNAG   ,    JNAGP  ,             106000   031A
      JNAUP  ,    JNAOPP ,    JNBMAX ,    JNRMAXP,    JNCB   ,             107000   031A
      JNC    ,    JNCMAX ,    JNCOL  ,    JNCM   ,    JNCMI  ,             108000   031A
      JNCN2  ,    JNCYC  ,    JNDEVG ,    JNDMAX ,    JNDMAXP,             109000   031A
      JNFORM ,    JNGROUP,    JNM1ST ,    JNMMAX ,    JNLB   ,             110000   031A
      JNLM   ,    JNMAT  ,    JNMATP ,    JNOGO  ,    JNPA   ,             111000   031A
      JNPAP  ,    JNPART ,    JNPHASE,    JNPCOL ,    JNPCOLP,             112000   031A
      JNPNOB ,    JNRA   ,    JNRFLG ,    JNRFLBP,    JNRING ,             113000   031A
      JNRMAX ,    JNRMAXP,    JNRSTOP,    JNSOREG,    JNSY   ,             114000   031A
           JNAZAD , JJAD , JIAO , JMAXR1 , JJAMAX ,                                  031A
           JNAZA , JNSAZA , JINDEX1 , JIII , JJ2 ,                                   031A
      JNSP   ,    JNUR   ,    JNRAIT ,        JNRB )                               031A
```

```
PROCEDURE SHRANDA(JIBASE,JRN);                                  130000    0314
INTEGER JIBASE;                                                 131000    0314
REAL JRN;                                                       132000    0314
BEGIN INTEGER A, B;                                             133000    0314
                                                              START OF SEGMENT ********** 0009
      A,[12118] ← JIBASE,[30118];                               134000    0000
      B,[121351 ← JIBASE,[13135];                               135000    0002
      JIBASE,[12136] ← A+B*JIBASE;                              136000    0004
      A ← 60;                                                   137000    0007
      A,[2712?1 ← JIBASE,[12127];                               138000    0008
      JRN ← A;                                                  139000    0010
      JRN ← JRN/134217728.0;                                    140000    0011
END SHRANDA;                                                    141000    0013
                                                              0009 IS 0017 LONG, NEXT SEG 0007
PROCEDURE SHSEARCH;                                             00044000  0314
BEGIN                                                           00045000  0314
INTEGER   JI,JJ,JK;                                                       0314
                                                              START OF SEGMENT ********** 0013
FORMAT FL23(/" BOUNDARY",I3," HAS BEEN INCORRECTLY IDENTIFIED."),  00050000  0000
                                                              START OF SEGMENT ********** 0011
FL37(/" POINT LIES ON BOUNDARY",I3),                            00051000  0000
FL85(/" SEARCH CYCLE THROUGH REGIONS IS NOT HANDLED PROPERLY."),  00052000  0000
FL95(/" CANNOT FIND REGION FOR POINT WITH COORDINATES R = ",S1,E10.3,  00053000  0000
 " H = ",S1,E10.3);                                                       0000
                                                              0011 IS 0058 LONG, NEXT SEG 0010
LIST LIST1(JNCH);                                               00054000  0000
LIST LIST2(JR,JH);                                              00056000  0005
LABEL L5,L10,L60,L50,L20,L25,L30,L35,L38,L40,L80,L90,L97,L01;    00057000  0012
L5: JNSV←0;                                                     00058000  0012
JNLR←JMPREG;                                                    00059000  0013
JNUB←JNRMAX;                                                    00060000  0014
L101: JK←JNLR;                                                  00061000  0015
```

```
00 BEGIN                                                                    00062000  0016
  JJ=SVNR[JK];                                                              00063000  0016
  JI=I;                                                                     00064000  0017
  DO BEGIN                                                                  00065000  0018
    JNCB=ABS(SVIB[JI,JK]);                                                  00066000  0018
    IF (XPR=(SVITY=E[JNCB[=1]))>0 THEN GO TO L30 ELSE IF XPH=0 THEN GO      00067000  0020
     TO L25;                                                                00068000  0023
    L20: WRITE(PRINT,FL23,LISTI);                                           00069000  0024
    JNHOA=JNHOA+1;                                                          00070000  0028
    GO TO L50;                                                              00071000  0030
    L25: JXH=SVCOEE[JNCH[=JH;                                               00072000  0030
    GO TO L35;                                                              00073000  0032
    L30: JXH=SVCOEE[JNCR)=JH;                                               00074000  0033
    L35: IF (XPH=(JXR))>0 THEN GO TO L40 ELSE IF XPR<0 THEN GO TO           00075000  0034
     L38;                                                                   00076000  0037
    WRITE(PRINT,FL37,LISTI);                                                00077000  0038
    JH=JH+JDELTA=JCOTH;                                                     00078000  0041
    JR=JR+JDELTA=JSITH=JCPH;                                                00079000  0043
    GO TO L5;                                                               00080000  0045
    L38: IF (XPH=(SVIB(J(,JK]))>0 THEN GO TO L60 ELSE IF XPH=0 THEN GO      00081000  0046
     TO L20 ELSE GO TO L50;                                                 00082000  0050
    L40: IF (XPH=(SVIB[JI,JK]))<0 THEN GO TO L40 ELSE IF XPH=0 THEN GO      00083000  0051
     TO L20;                                                                00084000  0055
    L50: END UNTIL (JI=(JI+1))>JJ;                                          00085000  0056
  JNCR=JK;                                                                  00086000  0058
  GO TO L0;                                                                 00087000  0059
  L60: END UNTIL (JK=(JK+1))>JNUB;                                          00088000  0059
IF (XPR=(JNSY))>0 THEN GO TO L90 ELSE IF XPH<0 THEN GO TO L80;              00089000  0062
JNSY=1;                                                                     00090000  0065
JNLB=I;                                                                     00091000  0066
JNUR=JNPREG;                                                                00092000  0066
GO TO L10;                                                                  00093000  0067
```

```
L80: WRITE(PRINT,FL85)}                                          00094000   0068
JWHOA←JWHOA+1}                                                   00095000   0071
GO TO L97}                                                       00096000   0072
L90: WRITE(PRINT,FL95,LIST2)}                                    00097000   0073
JWHOA←JWHOA+1}                                                   00098000   0077
L97: JNCR←0}                                                     00099000   0079
L6:  END}                                                        00100000   0079
                                                                 0010 IS 0083 LONG, NEXT SEG 0007
PROCEDURE SROSTRO}                                               00101000   0314
BEGIN                                                            00102000   0314
INTEGER      JJ,JK}                                                         0314
                                                                 START OF SEGMENT ********** 0012
COMMENT  THE FOLLOWING PROCEDURES ARE USED: SRSEARCH}            00109000   0000
FORMAT FL15(/" BOUNDARY",I3," HAS BEEN IDENTIFIED INCORRECTLY,"),  00110000   0000
                                                                 START OF SEGMENT ********** 0013
FL55(/" LOC =",I4,"  ICB =",I4,"  X =",S1,F10.3,"  BRAC =",S1,E10.3,  00111000   0000
 "  DIST =",S1,E10.3/" H =",S1,F10.3,"  R =",S1,E10.3,"  CURE(ICB) =",  00112000   0000
 S1,E10.3,"  ITYPE(ICB) =",I4),                                          0000
FL/5(/" COLLISION POINT IS WITHIN A DISTANCE OF 1.1 DELTA FROM BOUNDAR",  00114000   0000
"Y",I4,".  IT WAS MOVED OFF THE BOUNDARY.")}                     00115000   0000
                                                                 0013 IS 0072 LONG, NEXT SEG 0012
LIST LIST1(JICB)}                                                00116000   0000
LIST LIST2(JLOC,JICB,JX,JBRAC,JDIST,JH,JR,SVCOEE1J1CB1,SVITYPE1JICB)}  00117000   0005
LIST LIST3(JNCB)}                                                00118000   0021
LABEL L5,L40,L20,L30,L39,L36,L3A,L56,L0}                         00119000   0026
COMMENT  SUBROUTINE OSTRO}                                       00120000   0026
JNCB←0}                                                          00121000   0026
JJ1←1}                                                           00122000   0027
JLOC←105}                                                        00123000   0026
L5: JDIS:←JDLONG}                                                00124000   0028
JK←SVNB[JNCR1}                                                   00125000   0029
JJ←1}                                                            00126000   0030
```

```
DO BEGIN                                                                00127000  0031
  JICR=ABS(SVINT(JJ,JNCR1));                                            00128000  0031
  IF (XPR=(SVITYPE(JICR)-1))>0 THEN GO TO L30 ELSE IF XPR=0 THEN GO TO  00129000  0033
   L20;                                                                 00130000  0036
  WRITE(PRINT,L15,LIST1);                                               00131000  0037
  JWHOA=JWHOA+1;                                                        00132000  0041
  GO TO L0;                                                             00133000  0042
  L20: IF ABS(JCOTH1-SJSMVAL THEN GO TO L60;                            00134000  0042
  JX=(SVCOEF(JICR1-JH)/JCOTH1;                                          00135000  0044
  GO TO L39;                                                            00136000  0046
  L30: IF ABS(JSITH)-SJSMVAL THEN GO TO L60;                            00137000  0047
  JHRAC=(SVCOEF(JICH1)**2-(JH*JSPH1)**2);                               00138000  0048
  IF JHRAC<0 THEN GO TO L60;                                            00139000  0051
  IF (XPR=(SVCOEF(JICH1-JH))>0 THEN GO TO L38 ELSE IF XPR<0 THEN GO TO  00140000  0052
   L38;                                                                 00141000  0056
  JHPREG=JNCR1;                                                         00142000  0056
  SRSEARCH1;                                                            00143000  0057
  IF JFNHOR<JWHOA THEN GO TO L0 ELSE GO TO L5;                          00144000  0057
  L38: JX=(-JH*JCPH1-SQRT(JHRAC))/JSITH;                                00145000  0059
  GO TO L39;                                                            00146000  0063
  L38: JX=(-JH*JCPH1+SQRT(JHRAC))/JSITH;                                00147000  0065
  L39: IF JIDUMP<0 THEN GO TO L56;                                      00148000  0068
  WRITE(PRINT,L55,LIST2);                                               00149000  0070
  L54: IF JX<0 THEN GO TO L60;                                          00150000  0074
  IF JDIST<JX THEN GO TO L60;                                           00151000  0075
  JDIST=JX+JDELTA;                                                      00152000  0076
  JNCR=JICR;                                                            00153000  0077
  JJ1=JJ;                                                               00154000  0078
  L60: END UNTIL (JJ=(JJ+1))>JK;                                        00155000  007V
IF JDIST>1.1*JDELTA THEN GO TO L0;                                      00156000  0082
WRITE(PRINT,L75,LIST3);                                                 00157000  0084
JH=JH+JDELTA*JCOTH1;                                                    00158000  0087
```

```
JR+JR+JDELTA*JSITH*JCPHI;                                   00159000   0089
JMPREG+SVMPHI[JJ1,JNCR];                                    00160000   0091
SHSEARCH;                                                   00161000   0093
IF JNCR>0 THEN GO TO L5;                                    00162000   009A
L0:  END;                                                   00163000   0095
                                                          0012 IS 0101 LONG, NEXT SEG 0007
PROCEDURE SHDIFSCA;                                         00164000   031A
BEGIN                                                       00165000   031A
      INTEGER       J1 , JJAIL;                                        0314
                                                          START OF SEGMENT ********** 001A
COMMENT  THE FOLLOWING PROCEDURES ARE USED:  SHHANDA;       00171000   0000
FORMAT FL55(/                                               00172000   0000
                                                          START OF SEGMENT ********** 0015
 " THE COSINE VALUES FOR WHICH THE MIE SCATTERING PHASE FUNCTION "  00173000   0000
 "ARE INPUT ARE INCORRECT FOR MATERIAL",I3,".");            00174000   0000
                                                          0015 IS 0023 LONG, NEXT SEG 0014
LIST LIST1(JNCM);                                           00175000   0000
LABEL L5,L110,L150,L170,L200,L60,L52,L0;                    00176000   0005
SWITCH SWG01+L110,L150,L110,L150;                           00177000   0005
COMMENT  SUBROUTINE DIFSCA(POINT1);                         00178000   0011
IF JREFL50 THEN GO TO L5;                                   00179000   0011
JJAIL+SVJREFLT[JNRB];                                       00180000   0013
GO TO SWG01[JJAIL];                                         00181000   001A
L110: JPSCAT+1/0.2A318;                                     00182000   0016
GO TO L0;                                                   00183000   0018
L150: JNCYC+SVNMFANG[JNRB];                                 00184000   0020
J1+1;                                                       00185000   0021
DO BEGIN                                                    00186000   0021
  IF JCSA≥SVRFANG[J1,JNRB] THEN GO TO L170;                 00187000   0021
  END UNTIL (J1+(J1+11)>JNCYC);                             00188000   0024
  L170: IF JCSA = SVRFANG[J1,JNRB] THEN JPSCAT + SVPOR[J1,JNRB] ELSE   00188100   0026
      JPSCAT+SVPOR[J1-1,JNRB]+(SVPOR[J1,JNRB]-SVPOR[J1-1,JNRB])*(      00189000   0031
```

```
 JCSA=SVRFANGIJ1=I,JNRB))/(SVRFANGIJ1,JNRB1=SVRFANGIJ1=1,JNNB)))                00190000   0036
GO TO L01                                                                       00191000   00A3
L5: SRRANDA(JIBASE,JRN));                                                       00192000   0043
IF JRN>JRATLEE THEN GO TO L20;                                                  00193000   00A5
JPSCAT+(1+JCSA*JCSA)**.5*9868;                                                  00194000   0046
GO TO L01;                                                                      00195000   0048
L20: JNCYC+SVNDFCOSIJNCMI;                                                      00196000   0050
J1=1;                                                                           00197000   0051
DO BEGIN                                                                        00198000   0051
  IF JCSA2SVO1FCOSIJI,JNCMI THEN GO TO L60;                                     00199000   0051
  END UNTIL (J1+(J1+1))>JNCYC;                                                  00200000   005A
L52: WRITE(PRINT,FL55,LIST1));                                                  00201000   0056
JRMOA+JRMOA+1;                                                                  00202000   0060
GO TO L01                                                                       00203000   0062
   L60: IF JCSA = SVO1FCOSIJ1,JNCMI THEN JPSCAT + SVPOCOSIJI,JNCMI ELSE         00204000   0064
JPSCAT+SVPOCOSIJ1=1,JNCM)+(SVPOCOSIJI,JNCM1=SVPOCOSIJ1=1,JNCM))*(               00205000   0067
 JCSA=SVO1FCOSIJI=I,JNCM))/(SVO1FCOSIJ1,JNCM)=SVO1FCOSIJI=1,JNCM)));            00206000   0072
L01:  END;                                                                      00207000   0079
                                                                        OCIA IS 0083 LONG, NEXT SEG 0007
PROCEDURE SHOETECT;                                                             00208000   031A
BEGIN                                                                           00209000   031A
REAL    JCOD,  JSID;  INTEGER  JJ,JK,JL,JM;                                              031A
                                                                        START OF SEGMENT ********** 0018
      INTEGER     JLC, JI, JJ3, JJ2;                                                     0000
COMMENT  THE FOLLOWING PROCEDURES ARE USED:  SRRANDA,SROIFSCA;                  00220000   0000
F  RMAT FL513(/" AZIMUTHAL ANGLE ANG= ",S1,E11.3,                                       0000
                                                                        START OF SEGMENT ********** 0017
 "IS OUT OF INPUT AZIMUTHAL RANGE"),                                            00226000   0000
FL22(/" LOC =",IA," ALPHA =",S1,E10.3," BETA =",S1,F10.3," OIFH =",             00227000   0000
 S1,F10.3/" RRD2 =",S1,E10.3," RRDSQ =",S1,E10.3," SUMSQ =",S1,E10.3,           00228000   0000
 " ANG =",S1,E10.3/" NRING =",IA," J =",IA," K =",IA," CPT =",S1,               00229000   0000
 E10.3," SPT =",S1,E10.3/" OIFANG =",S1,E10.3," CPHIO =",S1,E10.3,              00230000   0000
```

```
" SPHID =",S1,E10.3," CPHRO =",S1,E10.3/" T =",S1,E10.3," CUTH =",          00231000    0000
S1,E10.3," TEHP =",S1,E10.3," S1TH =",S1,E10.3/" CPH1 =",S1,E10.3,          00232000    0000
" SPH1 =",S1,E10.3," H =",S1,E10.3," R =",S1,E10.3," RHOT =",S1,            00233000    0000
F10.3/" SUMDST =",S1,E10.3," HT =",S1,E10.3," DT =",S1,E10.3,               00234000    0000
" RN =",S1,E10.3);                                                                      0000
FL610(/" CAP= ",S1,F11.3," CARK= ",S1,F11.3," CPA= ",S1,E11.3," SPA= ",     00236000    0000
S1,E11.3/" S1D= ",S1,E11.3," COO= ",S1,E11.3," CPH10= ",S1,E11.3,           00237000    0000
" R2= ",S1,E11.3/" T= ",S1,E11.3," RD(J)= ",S1,E11.3);                      00238000    0000
FL2A5(/" CAP =",S1,E10.3," SAP =",S1,E10.3," APA =",S1,E10.3/               00239000    0000
" JA00 =",I3," CUAZ1 =",S1,E10.3," SDAZ1 =",S1,E10.3);                      00240000    0000
FL257(/" LOC =",I4," J =",I4," LA =",I4," LP =",I4," CSA =",S1,             002A1000    0000
E10.3," PSCAT =",S1,E10.3/" WAIT =",S1,E10.3," RHOT =",S1,E10.3,            00242000    0000
" NRING =",I4," CPA =",S1,E10.3/" RESULT =",S1,E10.3,                       00243000    0000
" FLUX(L,LP,LA) =",S1,E10.3," FLUD(J,NCR1) =",S1,E10.3/" NCH1 =",I4,        002AA000    0000
" RFLUX(J) =",S1,E10.3," REFL =",S1,E10.3," L =",I4);                       00245000    0000
                                                    0017 IS 0244 LONG, NEXT SEG 0018
LIST LIST1(JANG));                                                          002A6000    0000
LIST LIST2(JLOC,JALPHA,JBETA,J01FH,JRR02,JPROSQ,JSUHSQ,JANG,JNRING,         002A7000    0005
JJ,JK,JCPT,JSPT,J01FANG,JCPH10,JSPHID,JCPRRO,JT,JCUTH,JTEHP,JS1TH,          002A8000    0017
JCPH1,JSPH1,JH,JR,JRHOT,JSUMDST,JHT,JDT,JRN));                             002A9000    0032
LIST LIST3(JCAP,JCARK,JCPA,JSPA,JS10,JC00,JCPH10,JR2,JT,SVHD(JJ)));         00250000    00A7
LIST LIST4(JCAP,JSAP,JAPA,JJADO,JCDAZ1,JSDAZ1));                            00251000    0064
LIST LIST5(JLOC,JJ,JLA,JLP,JCSA,JPSCAT,JWAIT,JRHOT,JNRING,JCPA,JRESULT,S    00252000    0076
VFLUX(JLA,JLP,JL),SVFLUD(JNCH1,JJ1,JNCR1,SVRFLUX(JJ),JREFL,JL));            00253000    0091
BEGIN                                                                       00254000    0104
LABEL L240,L11,L503,L50A,L509,L510,L18,L17,L25,L100,L210,L650,L700,          00255000    0104
                                                    START OF SEGMENT ********** 0018
L217,LA50,L219,L230,L250,L255,L320,L0);                                     00256000    0000
COMMENT DETECT;                                                             00257000    0000
JALPHA=JS1TH2*JCPH12;                                                       00258000    0000
JBETA=JS1TH2*JSPH12;                                                        00259000    0001
JJ=1;                                                                       00260000    0002
```

```
DO BEGIN                                                                00261000  0003
  JO1FM=SVMO[JJ1-JM2]                                                   00262000  0003
  JRRO2=JR2*SVRO[JJ]**2                                                 00263000  0004
  JRROSQ=SVRO[JJ]*SVRO[JJ]+JM2*JM2                                      00264000  0006
  JSUNSQ=JRROSQ+JO1FM**2                                                00265000  0009
  JSOO=SQRT(SVRO[JJ]**2+(SVMO[JJ]-JMS)**2)                              00266000  0011
  JCOO=(SVMO[JJ]-JMS)/JSOO                                              00267000  0015
  JSIO=SVRO[JJ]/JSOO                                                    00268000  0017
  JNRING=SVNPNIO[JJ]                                                    00269000  0018
  JK=1                                                                  00270000  0019
  DO BEGIN                                                              00271000  0020
    SRRANDA(JIRAS1,JRN)                                                 00272000  0020
    IF SVA[1]<0 THEN GO TO L11                                          00273000  0021
    IF SVSAZA[JNSAZA]<3.1416 THEN GO TO L503                            00274000  0023
    JSAM=3.1416                                                         00275000  0024
    JRAT=.5                                                             00276000  0025
    GO TO L50A                                                          00277000  0026
    L503: JSAM=SVSAZA[JNSAZA]                                           00278000  0030
    JRAT=1                                                              00279000  0031
    L50A: JAZMAX=1-EXP(-SVA[1]*JSAM)                                    00280000  0031
    JANG=-LN(1-JRN*JAZMAX)/SVA[1]                                       00281000  0035
    JCMAIT=(JRAIT*JAZMAX/(SVA[1]*EXP(-SVA[1]*JANG)))/JRAT               00282000  0038
    SRRANDA(JIRAS2,JRN)                                                 00283000  0043
    IF JRN<JRAT THEN GO TO L509                                         00284000  0044
    JANG=6.2832-JANG                                                    00285000  0045
    L509: J1=2                                                          00286000  0046
    DO BEGIN                                                            00287000  0047
      IF JANG<SVSAZA[J1] THEN GO TO L510                                00288000  0047
      END UNTIL (J1=(J1+1))>JNSAZA                                      00289000  0049
    WRITE(PRINT,FL913,LIST1)                                            00290000  0051
    JMMDA=JMMDA+1                                                       00291000  0055
    GO TO L2A0                                                          00292000  0056
```

```
L510: JPAZ=SVPAZA[JI-1]+(SVPAZA[JI]-SVPAZA[JI-1])*(JANG-SVSAZA[      00293000   0081
 JI-1])/(SVSAZA[JI]-SVSAZA[JI-1]);                                   00294000   0082
JCWAIT=JCWAIT*JPAZ;                                                  00295000   0083
GO TO L180;                                                          00296000   0084
L11: JI=2;                                                           00297000   0085
DO BEGIN                                                             00298000   0086
  IF JRN<SVPAZA[JI] THEN GO TO L17;                                  00299000   0087
  END UNTIL (JI=(JI+1))>JNSAZA;                                      00300000   0088
L17: JANG=SVSAZA[JI-1]+(SVSAZA[JI]-SVSAZA[JI-1])*(JRN-SVP   A1      00301000   0089
 JI-1])/(SVPAZA[JI]-SVPAZA[JI-1]);                                   00302000   0090
JCWAIT=JWAIT;                                                        00303000   0091
L180: JOIFANG=SVAZO[JJI-JANG;                                        00304000   0092
JCPT=COS(JOIFANG);                                                   00305000   0093
JSPT=SIN(JOIFANG);                                                   00306000   0094
JCPH1O=JCPT*JCOAZI+JSPI*JSOAZI;                                      00307000   0095
JSPH1O=JSPT*JCOAZI-JCPI*JSOAZI;                                      00308000   0096
JCPHRO=JCPH1O*JRRO2;                                                 00309000   0097
IF (JSUMSQ-JCPHRO) < 0 THEN GO TO L260 ;                             00309100   0098
JT=SQRT(JSUMSQ-JCPHRO);                                              00310000   0099
IF JT≤JDMIN THEN GO TO L260;                                         00311000   0100
JCOTH=JOIFH/JT;                                                      00312000   0101
JTEMP=SQRT(JRRQSQ-JCPHRO);                                           00313000   0102
JSITH=JTEMP/JT;                                                      00314000   0103
JLOC=90;                                                             00315000   0104
IF JIOUMPSO THEN GO TO L25;                                          00316000   0105
WRITE(PRINT,FL22,LIST2);                                             00317000   0106
L25: IF ABS(JCOTH)>JSMVAL THEN GO TO L100;                           00318000   0107
JRHOT=JT*(SVTAU[JJHT]-SVTAU[JJHR])/(SVHV[JJHT]-SVHV[JJHR]);          00319000   0108
GO TO L210;                                                          00320000   0109
L100: JRHOT=(SVTAUHO[JJI-JTAUH2)/JCOTH;                              00321000   0110
L210: JCSA=(JALPHA*(SVRO[JJ]*JCPH1O-JH2)+JBETA*(SVRO[JJ]*JSPH1O)+    00322000   0111
 JCOTH2*JOIFH)/JT;                                                   00323000   0112
```

```
SROIFSCA)                                                    00324000   0126
IF JERRORS<JNMDA THEN GO TO L~)                              00325000   0127
JREFSUL?+' JC*AI?*JPSCA?*EXP(-JHMD?))/((JNRING)*J1*2))       00326000   0128
JCARK+(SVRD[JJ]-JR2*JCPHIO)/JT)                              00327000   0132
JCPA+JSID*JCARK+JCOTH*JCOD)                                  00328000   0135
IF JCPA*2 ≥ 1 THEN JSPA + 0 ELSE                             00328100   0137
JSPA+SQRT(1-JCPA*2))                                         00329000   0140
IF JSPA = 0 THEN JCAP + 0.9999 ELSE                          00329100   0144
IF JCOD ¬ 0 THEN JCAP + JCOTH / SQRT(1 + JCARK*2) ELSE       00329200   0146
JCAP+(JCARK+JSID*JCPA)/(-1*JCOD*JSPA))                       00330000   0153
IF ABS(JCAP)≤1 THEN GO TO L650)                              00331000   0158
WRITE(PRINT,FL610,LIST3))                                    00332000   0160
GO TO L260)                                                  00333000   0163
L650: JSAP+SQRT(1-JCAP*2))                                   00334000   0164
JSAP+TSIGN(JSAP,JSPHIO))                                     00335000   0167
IF JCAP = 0 THEN JAPA + SIGN(JSAP)*1.570796 ELSE             00335100   0169
BEGIN                                                        00335200   0173
JARG+JSAP/JCAP)                                              00336000   0176
JAPA+ARCTAN(JARG))                                           00337000   0177
END )                                                        00337100   0178
IF JCAP≥0 THEN GO TO L700)                                   00338000   0178
JSAM+3.1416)                                                 00338500   0180
JAPA+JAPA+TSIGN(JSAM,JSAP))                                  00339000   0180
L700: IF JAPA≥0 THEN GO TO L217)                             00340000   0182
IF SVCAZA[JNAZA]>10) THEN GO TO L650)                        00341000   0184
JAPA+ABS(JAPA))                                              00342000   0185
GO TO L217)                                                  00343000   0186
L650: JAPA+JAPA+6.28318)                                     00344000   0190
L217: JAPA+JAPA*57.295/19)                                   00345000   0191
JL+1)                                                        00346000   0193
DO BEGIN                                                     00347000   0194
  IF JAPA≤SVCAZA[JL] THEN GO TO L219)                        00348000   0194
```

```
      END UNTIL (JL:=(JL+1))>JNAZA;                    00349000   0195
    L219: JLAZ:=JL;                                    00350000   0197
    JJA00:=JLAZ+JNAZA*(JJ-1);                          00351000   0198
    JL:=1;                                             00352000   0201
    DO BEGIN                                           00353000   0201
      IF SVCIPA[JL]≤JCPA THEN GO TO L230;              00354000   0201
      END UNTIL (JL:=(JL+1))>JNPA;                     00355000   0203
    L230: JLA:=JL;                                     00356000   0205
    IF JIOUMPSO THEN GO TO L250;                       00357000   0206
    WRITE(PRINT,FL205,LIST4);                          00358000   0208
    L250: JLP:=JNREFL;                                 00359000   0211
    SVFLUX[JLA,JLP,JJA00]:=SVFLUX[JLA,JLP,JJA00]+JRESULT;  00360000   0212
    SVFLUO[JNCR2,JJJ]:=SVFLUO[JNCR2,JJJ]+JRESULT;      00361000   0217
    IF JREFLSO THEN GO TO L255;                        00362000   0221
    SVRFLUX[JJJ]:=SVRFLUX[JJJ]+JRESULT;                00363000   0222
    SVRUO[JLP,JJJ]:=SVRUO[JLP,JJJ+1]+JRESULT;          00364000   0224
    L255: JM:=1;                                       00365000   0228
    DO BEGIN                                           00366000   0228
      IF SVINCOL[JM]>JNCOL THEN GO TO L320;            00367000   0228
      END UNTIL (JM:=(JM+1))>JNPCOL;                   00368000   0230
    L320: JLC:=JM;                                     00369000   0232
    SVAFLUX[JLC,JJJ]:=SVAFLUX[JLC,JJJ]+JRESULT;        00370000   0233
    JLOC:=1101;                                        00371000   0237
    IF JIOUMPSO  HEN GO TO L260;                       00372000   0238
    JL:=JJA00;                                         00373000   0239
    WRITE(PRINT,FL257,LIST5);                          00374000   0240
    L240: END UNTIL (JK:=(JK+1))>JNRING;               00375000   02A3
  END UNTIL (JJ:=(JJ+1))>JNUMAX;                       00376000   02A6
L01:  END END;                                         00377000   02A8

                                                  001P IS 0252 LONG, NEXT SEG 0010

                                                  0010 IS 0111 LONG, NEXT SEG 0007

PROCEDURE SHANSWER;                                    00378000   031A
```

```
BEGIN                                                                00379000    0314
OWN INTEGER OXII                                                     00380000    031A
                                                                   START OF SEGMENT ********** 0019
      INTEGER  JJN, JJJ , JNCARD                                                   0000
REAL  JFGROUP,JFNHMAXJ     INTEGER  JI,JJ,JK,JN,JM ;                              0000
FORMAT FL110(" RADIATION RESEARCH ASSOCIATES LITE"I PROBLEM",I10),    00392000    0000
                                                                   START OF SEGMENT ********** 0020
FL120(/" HISTORY TERMINATION COUNTERS."),                            00393000    0000
FL130(/" ",I9,                                                       00394000    0000
 " HISTORIES WERE TERMINATED WHEN THE COLLISION NUMBER EXCEEDED",I6,"."/     00395000    0000
I10," HISTORIES WERE TERMINATED BY THE REGION IMPORTANCE PARAMETERS,"/       00396000    0000
 I10," HISTORIES WERE TERMINATED BY MINIMUM WEIGHT CUTOFF,"/I10,             00397000    0000
 " HISTORIES WERE TERMINATED AFTER  MAXIMUM NUMBER OF REFLECTIONS",",",      00398000    0000
FL135(/" ",I9," COLLISIONS OCCURRED."),                              00399000    0000
FL150(/                                                              00400000    0000
 " PARTICLES TERMINATED IN EACH REGION BY REGION IMPORTANCE PARAM",          00401000    0000
 "ETERS."),                                                                  00402000    0000
FL160"/                                                              00403000    0000
 " REGION HISTORIES   REGION HISTORIES   REGION HISTORIES   REGIO",          00404000    0000
 "N HISTORIES"/                                                              00405000    0000
 "        TERMINATED          TERMINATED          TERMINATED       ",        00406000    0000
 "  TERMINATED"),                                                            00407000    0000
FL170(" ",IA,I9,I10,I9,I10,I9,I10,I9),                               00408000    0000
FL190(/                                                              00409000    0000
 "  SCATTERED LIGHT INTENSITY VERSUS ANGLE AND NUMBER OF  REFLECT",          00410000    0000
 "IONS FROM SURFACE ONE."),                                                  00411000    0000
FL191(/X10," AZIMUTHAL RANGE =  ",SI,E10.3," TO  ",SI,E10.3),        00412000    0000
FL200(/"       SOURCE HEIGHT H=",SI,F10.3,                                              0000
 ",  DETECTOR COORDINATES  HO=",SI,E10.3," HO=,",SI,E10.3),           00414000    0000
FL210(" ANGLE",X3,"COLLISION"),                                      00415000    0000
FL250(" (COSINE)",I8,6(X9,I2)),                                      00416000    0000
FL262(" (COSINE)    TOTAL"),                                         00417000    0000
```

```
FL264(" ",X23,"TOTAL"),                                                  00418000    0000
FL266(" ",X34,"TOTAL"),                                                  00419000    0000
FL268(" ",X45,"TOTAL"),                                                  00420000    0000
FL270(" ",X56,"TOTAL"),                                                  00421000    0000
FL272(" ",X67,"TOTAL"),                                                  00422000    0000
FL274(" ",X78,"TOTAL"),                                                  00423000    0000
                                                                  0020 IS 0261 LONG, NEXT SEG 0019
                                                                  START OF SEGMENT ********** 0021
FL280(" ",R7.4,X1,S1,7E11.3),                                                        0000
FL300(/"  TOTAL  ",S1,7E11.3),                                                       0000
FL450(/                                                                  00426000    0000
 "                    SCATTERED LIGHT INTENSITY VERSUS REGION OF ",      00427000    0000
 "SCATTER"),                                                             00428000    0000
FL460(/"  REGION    ",X30,"DETECTOR"),                                   00429000    0000
FL485(/"                 01"),                                           00430000    0000
FL495(/"                 01          02"),                               00431000    0000
FL505(/"                 01          02          03"),                   00432000    0000
FL515(/"                 01          02          03          04"),       00433000    0000
FL525(/"                 01          02          03          04          05"),  00434000  0000
FL535(/                                                                  00435000    0000
 "               01          02          03          04          05  ",  00436000    0000
 "       06"),                                                           00437000    0000
FL545(/                                                                  00438000    0000
 "               01          02          03          04          05  ",  00439000    0000
 "       06          07"),                                               00440000    0000
FL560("    ",I2,X3,S1,7E11.3),                                                       0000
FL580(/"  TOTAL  ",S1,7E11.3),                                                      0000
FL605(/"                 08"),                                           00443000    0000
FL615(/"                 08          09"),                               00444000    0000
FL625(/"                 08          09          10"),                   00445000    0000
FL680(" LIGHT SCATTERED FROM REFLECTION SURFACES TO EACH DETECTOR."),    00446000    0000
FL690(/" NO OF REFLECTIONS                          DETECTOR "),         00447000    0000
```

```
FL705(/" ",X12,5(X9,I2)),                                          00AA8000   0000
FL710("        ",I2,X9,$I,5E11.3),                                            0000
FL720(/"     TOTAL         ",$I,5E11.3),                                     0000
FL735(7H8,A,X10,IA,I2,I2,I4," ACC"),                               00A91000   0000
FL745($I,AE11.3,I4,I2,I2,I4," ACC"),                                          0000
FL747($I,3F11.3,X33,IA,I2,I2,I2," ACC"));                                     0000
                                                                   0021 IS 0261 LONG, NEXT SEG 0019
LIST LIST1(JNPH08));                                               00A5A000   0000
LIST LIST2(JMAXCOL,JNCMAX,JNRSTOP,JNWAIT,JNMAXR));                 00A55000   0005
LIST LIST3(JNOGO));                                                00A56000   0015
LIST LIST4(FOR OX1=1 STEP 1 UNTIL JNRMAX DO (OX1,SVNRIC[OX1]));    00A57000   0020
LIST LIST5(JCAZAD,SVCCAZA[JIAD1]));                                00A58000   0031
LIST LIST6(JMS,SVH0[JJJ],SVH01JJJ]));                              00A59000   0038
LIST LIST7(FOR OX1=JKA1 STEP 1 UNTIL JKA2 DO SVIINEE[OX1]));       00A60000   0046
LIST LIST8(SVC1PA[JN],FOR OX1=JKA1 STEP 1 UNTIL JKA2 DO SVFLUX1JN,OX1,   00A61000   0055
 JJA001));                                                         00A62000   0061
LIST LIST9(FOR OX1=JKA1 STEP 1 UNTIL A2 DO SVTFLUX[OX1,JJA001));   00A63000   0077
LIST LIST10(SVNNEG[JI],FOR OX1=1 ST  1 UNTIL JNFORM DO SVFLL0[JI,OX1]));  00A64000   0077
LIST LIST11(FOR OX1=1 STEP 1 UNTIL JNFORM DO SVFLUR[OX1]));        00A65000   0088
LIST LIST12(SVNNEG[JI],FOR OX1=A STEP 1 UNTIL JNFORM DO SVFLU0[JI,OX1]));  00A66000   0097
LIST LIST13(FOR OX1=A STEP 1 UNTIL JNFORM DO SVFLUR[OX1]));        00A67000   0108
LIST LIST14(FOR OX1=JKA1 STEP 1 UNTIL JKA2 DO SVNDET[OX1]));       00A68000   0117
LIST LIST15(SV11REE[JJJ],FOR OX1=JKA1 STEP 1 UNTIL JKA2 DO SVNGO1  00A69000   0126
 JJJ,OX1]));                                                       00A70000   0130
LIST LIST16(FOR OX1=JKA1 STEP 1 UNTIL JKA2 DO SVNFLUX[OX1]));      00A71000   0137
LIST LIST17(FOR OX1=JKA1 STEP 1 UNTIL JKA2 DO SVANG[OX1],JNPHOR,JIAD,  00A72000   0146
 JJD,JNCARD));                                                     00A73000   0155
LIST LIST18(FOR OX1=JKA1 STEP 1 UNTIL JKA2 DO SVFLUX[JJN,OX1,JJJI,  00A74000   0160
 JNPROR,JIAD,JJD,JNCARD));                                         00A75000   0168
BEGIN                                                              00A76000   0176
LABEL L998,L180,L185,L280,L281,L275,L283,L285,L287,L289,L271,L273,  00A77000   0176
                                                                   START OF SEGMENT ********** 0022
```

```
 L14,L430,L440,L480,L490,L500,L510,L520,L530,L540,L600,L610,L620,          00478000  0000
 L550,L663,L650,L670,L700,L770,L730,L0;                                    00479000  0000
SWITCH SWG01:=L261,L263,L265,L267,L269,L271,L273,L275;                     00480000  0000
SWITCH SWG02:=L480,L490,L500,L510,L520,L530,L540,L600,L610,L620;           00481000  0007
COMMENT  SUBROUTINE ANSWER;                                                00482000  0016
JFNMMAX:=JNMMAX;                                                           00483000  0016
JFGROUP:=JNGROUP;                                                          00484000  0017
JJADMAX:=JNUMAXM*JNAZA;                                                    00485000  0018
JLST:=JM#XR+1;                                                             00486000  0019
JJ:=1;                                                                     00487000  0021
DO BEGIN                                                                   00488000  0021
  J1:=1;                                                                   00489000  0021
  DO BEGIN                                                                 00490000  0022
    JK:=1;                                                                 00491000  0022
    DO BEGIN                                                               00492000  0023
      SVFLUX[JK,JI,JJ]:=SVFLUX[JK,J[,JJ]/JFNMMAX;                          00493000  0023
      SVFLUX[JK,JLST,JJ]:=SVFLUX[JK,JLST,JJ]+SVFLUX[JK,JI,JJ];             00494000  0026
      SVTFLUX[J1,JJ]:=SVTFLUX[J1,JJ]+SVFLUX[JK,JI,JJ];                     00495000  0035
      END UNTIL (JK:=(JK+1))>JNPA;                                         00496000  0040
    SVTFLUX[JLST,JJ]:=SVTFLUX[JLST,JJ]+SVTFLUX[JI,JJ];                     00497000  0042
    SVTIREE[J1]:=JI-1;                                                     00498000  0047
    END UNTIL (J1:=(JI+1))>JM#XM END UNTIL (JJ:=(JJ+1))>JJADMAX;           00499000  0048
JJ:=1;                                                                     00500000  0053
DO BEGIN                                                                   00501000  0054
  JM:=1;                                                                   00502000  0054
  DO BEGIN                                                                 00503000  0054
    SVFLUO[JM,JJ]:=SVFLUO[JM,JJ]/JFNMMAX;                                 00504000  0054
    SVFLUR[JJ]:=SVFLUR[JJ]+SVFLUO[JM,JJ];                                  00505000  0058
    END UNTIL (JM:=(JM+1))>JNMMAX;                                         00506000  0061
  SVRFLUX[JJ]:=SVRFLUX[JJ]/JFNMMAX;                                        00507000  0063
  JI:=1;                                                                   00508000  0065
  DO BEGIN                                                                 00509000  0066
```

```
    SVRUD[JI,JJJ]*SVRUD[JI,JJJ]/JENHMAX;                    00510000  0066
    END UNTIL (JI*(JI+1))>JMAXH  END UNTIL (JJ*(JJ+1))>JNDMAX;  00511000  0069
COMMENT  SUBROUTINE RESULT;                                  00512000  007A
WRITE(PRINT[PAGE]);                                          00513000  0074
WRITE(PRINT,FL110,LIST1);                                    00514000  0077
WRITE(PRINT,FL120);                                          00515000  0081
WRITE(PRINT,FL130,LIST2);                                    00516000  008A
WRITE(PRINT,FL135,LIST3);                                    00517000  0086
IF JNHSTOPSO THEN GO TO L9YY;                                00518000  0092
WRITE(PRINT,FL150);                                          00519000  0093
WRITE(PRINT,FL160);                                          00520000  0097
WRITE(PRINT,FL170,LIST4);                                    00521000  0100
L9Y8; JNOUNT+0;                                              00522000  0104
JJ+1;                                                        00523000  0105
DO BEGIN                                                     00524000  0106
  SVCCAZA[JJ]*SVCAZA[JJ] END UNTIL (JJ*(JJ+1))>JNAZA;        00525000  0108
JNAZAO+JNAZA;                                                00526000  0110
L180; JJ+1;                                                  00527000  0111
DO BEGIN                                                     00528000  0111
  JJAO+(JJ-1)*JNAZAO;                                        00529000  0111
  JCAZAU+0;                                                  00530000  0113
  JIAO+1;                                                    00531000  0114
  DO BEGIN                                                   00532000  0115
    JJAOO+JJAO+JIAO;                                         00533000  0115
    JKA2+0;                                                  00534000  0116
    JKA3+0;                                                  00535000  0117
    L185; WRITE(PRINT[PAGE,]);                               00536000  0117
          WRITE(PRINT,FL110,LIST1);                          00537000  0121
    WRITE(PRINT,FL190);                                      00538000  0125
    WRITE(PRINT,FL191,LIST5);                                00539000  0128
    WRITE(PRINT,FL200,LIST6);                                00540000  0132
    WRITE(PRINT,FL210);                                      005A1000  0136
```

```
JKA1=JKA2+1;                                                    00542000  0139
JKA2=JKA1+6;                                                    00543000  0140
IF JKA2>JMAXR THEN GO TO L240;                                  00544000  0142
JKA3=1;                                                         00545000  0143
JKA2=JMAXR;                                                     00546000  0144
IF JKA1>JMAXR THEN GO TO L261;                                  00547000  0144
L240: WRITE(PRINT,FL250,LIST7);                                 00548000  0146
IF JKA3=0 THEN GO TO L275;                                      00549000  0149
JKA2=JKA2+1;                                                    00550000  0151
JKA4=JKA2-JKA1+1;                                               00551000  0152
GO TO SWGO1(JKA4);                                              00552000  0154
L261: WRITE(PRINT,FL262);                                       00553000  0156
GO TO L275;                                                     00554000  0159
L263: WRITE(PRINT    ,FL264);                                             0161
GO TO L275;                                                     00555000  0163
L265: WRITE(PRINT    ,FL266);                                             0164
GO TO L275;                                                     00556000  0167
L267: WRITE(PRINT    ,FL268);                                             0168
GO TO L275;                                                     00560000  0171
L269: WRITE(PRINT    ,FL270);                                             0172
GO TO L275;                                                     00562000  0175
L271: WRITE(PRINT    ,FL272);                                             0176
GO TO L275;                                                     00564000  0179
L273: WRITE(PRINT    ,FL274);                                             0180
L275: JN=1;                                                     00566000  0183
DO BEGIN                                                        00567000  0184
  WRITE(PRINT,FL280,LIST8);                                     00568000  0184
  END UNTIL (JN=(JN+1))>JNPA;                                   00569000  0188
WRITE(PRINT,FL300,LIST9);                                       00570000  0190
IF JKA3=0 THEN GO TO L165;                                      00571000  0194
JCAZA0=SVCC#ZAT(JIA0);                                          00572000  0195
END UNTIL (J1A0=(J1A0+1))>JNAZA0 END UNTIL (JJ=(JJ+1))>JNDMAX;  00573000  0196
```

```
IF JKOUNT#0 THEN GO TO L18;                                00574000   0201
IF JNDM#X>7 THEN GO TO L430;                               00575000   0202
JNFORM#=JNDM#X;                                            00576000   0203
GO TO L440;                                                00577000   0204
L430: JNFORM#=7;                                           00578000   0205
L440: WRITE(PRINT#PAGE));                                  00579000   0205
      WRITE(PRINT,FL110,LIST(1));                          00580000   0209
WRITE(PRINT,FL450);                                        00581000   0213
WRITE(PRINT,FL460);                                        00582000   0216
GO TO SWGO2(JNFORM);                                       00583000   0220
L480: WRITE(PRINT,FL485);                                  00584000   0222
GO TO L550;                                                00585000   0225
L490: WRITE(PRINT,FL495);                                  00586000   0226
GO TO L550;                                                00587000   0229
L500: WRITE(PRINT,FL505);                                  00588000   0230
GO TO L550;                                                00589000   0233
L510: WRITE(PRINT,FL515);                                  00590000   0234
GO TO L550;                                                00591000   0237
L520: WRITE(PRINT,FL525);                                  00592000   0238
GO TO L550;                                                00593000   0241
L530: WRITE(PRINT,FL535);                                  00594000   0242
GO TO L550;                                                00595000   0245
L540: WRITE(PRINT,FL545);                                  00596000   0246
L550: JI=1;                                                00597000   0249
DO BEGIN                                                   00598000   0250
  WRITE(PRINT,FL560,LIST(0));                              00599000   0250
  END UNTIL (JI=(JI+1))>JNDM#X;                            00600000   0254
WRITE(PRINT,FL580,LIST(I));                                00601000   0256
IF JNDM#X<JNFORM THEN GO TO L663;                          00602000   0260
JNFORM=JNDM#X;                                             00603000   0261
GO TO L440;                                                00604000   0262
L600: WRITE(PRINT,FL605);                                  00605000   0263
```

```
GO TO L650;                                                    00606000  0266
L610: WRITE(PRINT,FL615);                                      00607000  0267
GO TO L650;                                                    00608000  0270
L620: WRITE(PRINT,FL625);                                      00609000  0271
L650: JI+1;                                                    00610000  0274
DO BEGIN                                                       00611000  0275
  WRITE(PRINT,FL560,LIST12);                                   00612000  0275
  END UNTIL (JI+(JI+1))>JNMMAX;                                00613000  0277
WRITE(PRINT,FL580,LIST13);                                     00614000  0281
L663: JJJ+1;                                                   00615000  0285
DO BEGIN                                                       00616000  0286
  SVNDET[JJJ]+JJJ END UNTIL (JJJ+(JJJ+1))>JNOMAX;              00617000  0286
JKA2+0;                                                        00618000  0290
L670: WRITE(PRINT(PAGE));                                      00619000  0291
      WRITE(PRINT,FL680);                                      00620000  0294
WRITE(PRINT,FL690);                                            00621000  0297
JKA1+JKA2+1;                                                   00622000  0301
JKA2+JKA1+0;                                                   00623000  0302
IF JKA2≤JNOMAX THEN GO TO L700;                                00624000  0303
JKA2+JNOMAX;                                                   00625000  0305
L700: WRITE(PRINT,FL705,LIST14);                               00626000  0305
JJJ+2;                                                         00627000  0309
DO BEGIN                                                       00628000  0310
  WRITE(PRINT,FL710,LIST15);                                   00629000  0310
  END UNTIL (JJJ+(JJJ+1))>JMAXR;                               00630000  0314
WRITE(PRINT,FL720,LIST16);                                     00631000  0315
IF JKA2<JNOMAX THEN GO TO L670;                                00632000  0320
SVANG(1)+1;                                                    00633000  0321
JJJ+1;                                                         00634000  0322
DO BEGIN                                                       00635000  0323
  SVANG(JJJ+1)+SVCIRA(JJJ) END UNTIL (JJJ+(JJJ+1))>JNPA;       00636000  0323
LIA: JJJ+1;                                                    00637000  0327
```

```
00 BEGIN                                                                    00638000  0328
  JJAD=(JJ0=1)=JNAZA0;                                                      00639000  0328
  JI40=1;                                                                   00640000  0330
  DO BEGIN                                                                  00641000  0331
    JJJ=JJAD=JI40;                                                          00642000  0331
    JNCARD=0;                                                               00643000  0332
    JKA2=0;                                                                 00644000  0333
    L730: JKA1=JKA2=1;                                                      00645000  0334
    JKA2=JKA1=6;                                                            00646000  0335
    JNCARD=JNCARD=1;                                                        00647000  0336
    WRITE(PUNCH,FL735,LIST1A);                                              00648000  0337
    IF JKA2=JNPA<1 THEN GO TO L730;                                         00649000  0341
    JJN=1;                                                                  00650000  0343
    DO BEGIN                                                                00651000  0344
      JKA2=0;                                                               00652000  0344
      JKA1=JKA2=1;                                                          00653000  0346
      JKA2=JKA1=5;                                                          00654000  0346
      JNCARD=JNCARD=1;                                                      00655000  0347
      WRITE(PUNCH,FL745,LIST18);                                            00656000  0346
      IF JKA2=JMAXR THEN GO TO L770;                                        00657000  0352
      JKA1=JKA2=1;                                                          00658000  0353
      JKA2=9;                                                               00659000  0354
      JNCARD=JNCARD=1;                                                      00660000  0355
      WRITE(PUNCH,FL747,LIST18);                                            00661000  0356
      L770: END UNTIL (JJN=(JJN=1))>JNPA END UNTIL (JIAD=(JI40=1));         00662000  0360
    JNAZAD END UNTIL (JJD=(JJ0=1))>JNOMAX;                                  00663000  0364
  IF JNAZAD=1 THEN GO TO L01;                                               00664000  0367
  JMAXR1=JMAXR=1;                                                           00665000  0369
  JI=1;                                                                     00666000  0370
  00 BEGIN                                                                  00667000  0371
    JK=1;                                                                   00668000  0371
    DO BEGIN                                                                00669000  0371
```

```
    JJ+1;                                                          00A70000   0371
    DO BEGIN                                                       00A71000   0372
      JJAD+(JJ-1)*JNAZAD;                                          00A72000   0372
      JR1+0;                                                       00A73000   0374
      JIAD+1;                                                      00A74000   0375
      DO BEGIN                                                     00A75000   0375
        JJADD+JJAD+JIAD;                                           00A76000   0375
        SVFLUX[JK,JI,JJ]+SVFLUX[JK,JI,JJADD]+JR1;                  00A77000   0377
        JR1+SVFLUX[JK,JI,JJ];                                      00A78000   0382
        END UNTIL (JIAD+(JIAD+1))>JNAZAD END UNTIL (JJ+(JJ+1))>JNDMAX  00A79000   0384
     END UNTIL (JK+(JK+1))>JNPA END UNTIL (JI+(JI+1))>JMAXR1;      00A80000   0386
JI+1;                                                              00A81000   0393
DO BEGIN                                                           00A82000   0394
  JJ+1;                                                            00A83000   0394
  DO BEGIN                                                         00A84000   0395
    JJAD+(JJ-1)*JNAZAD;                                            00A85000   0395
    JR2+0;                                                         00A86000   0396
    JIAD+1;                                                        00A87000   0397
    DO BEGIN                                                       00A88000   0398
      JJADD+JJAD+JIAD;                                             00A89000   0398
      SVTFLUX[JI,JJ]+SVTFLUX[JI,JJADD]+JR2;                        00A90000   0399
      JR2+SVTFLUX[JI,JJ];                                          00A91000   0403
      END UNTIL (JIAD+(JIAD+1))>JNAZAD END UNTIL (JJ+(JJ+1))>JNDMAX END  00A92000   0404
 UNTIL (JI+(JI+1))>JMAXR1;                                         00A93000   0A09
JNAZAD+1;                                                          00A9A000   0A11
SVCCAZA[1]+SVCAZA[JNAZA];                                          00A95000   0A12
JRCUNT+1;                                                          00A9A000   0A13
GO TO L1A0;                                                        00A97000   0A14
L0:  END END;                                                      00A98000   0A15
                                                  0022 IS 0A16 LONG, NEXT SEG 0019
                                                  0019 IS 0126 LONG, NEXT SEG 0007
PROCEDURE SRAVRAGE;                                                00A99000   031A
```

```
BEGIN                                                                   00700000    0314
INTEGER  DX1,J1,JJ,JK ,JINDX I                                                      0314
                                                                    START OF SEGMENT ********** 0023
REAL JFPART,JFGROUP1                                                                0000
FORMAT FL110(" ",X99,"FLUXES FOR DEVIATION GROUP",I3,",");                00708000  0000
                                                                    START OF SEGMENT ********** 0024
FL120(/" COLLISIONS",X30,"DETECTOR"),                                     00709000  0000
FL145(/"                01"),                                             00710000  0000
FL155(/"                01          02"),                                 00711000  0000
FL165(/"                01          02          03"),                     00712000  0000
FL175(/"                01          02          03          04"),         00713000  0000
FL185(/"                01          02          03          04          05"), 00714000  0000
FL195(/                                                                   00715000  0000
 "              01          02          03          04          05  ",     00716000  0000
 "       06"),                                                            00717000  0000
FL205(/                                                                   00718000  0000
 "              01          02          03          04          05  ",     00719000  0000
 "       06          07"),                                                00720000  0000
FL220("    ",I2,X3,51,7E11.3),                                                      0000
FL230(/"  TOTAL  ",51,7E11.3),                                                     0000
FL265(/"                08"),                                             00723000  0000
FL275(/"                08          09"),                                 00724000  0000
FL285(/"                08          09          10"),                     00725000  0000
FL320(/" BASE FOR RANDOM NUMBER GENERATOR IS",113),                       00726000  0000
FL400(" ",X11,                                                            00727000  0000
 " SCATTERED INTENSITIES VERSUS DETECTOR AND COLLISION NUMBER,"),          00728000  0000
FL460(" ",X11,                                                            00729000  0000
 " INTENSITY DEVIATIONS VERSUS DETECTOR AND COLLISION NUMBER,"));         00730000  0000
                                                                    IS 0196 LONG, NEXT SEG 0023
LIST LIST1(JNDEVG));                                                      00731000  0000
LIST LIST2(SVINCOLI,J1,FOR DX1=1 STEP 1 UNTIL JNFORM DO ,VARLUX1,J1,DX1));00732000  0005
 1                                                                        00733000  0010
```

```
LIST LIST3(FOR OXI:=1 STEP 1 UNTIL JNFORM DO SVSTFLUX[OXI])$                    00734000   0016
LIST LIST4(SVINCOL[JI],FOR OXI:=0 STEP 1 UNTIL JNOMAX DO SVAFLUX[JI,OXI])       00735000   0025
 $                                                                              00736000   0030
LIST LIST5(FOR OXI:=0 STEP 1 UNTIL JNOMAX DO SVSTFLUX[OXI])$                    00737000   0036
LIST LIST6(JIBASE)$                                                             00738000   00A5
LABEL L115,L125,L130,L140,L150,L160,L170,L180,L190,L200,L210,L310,               00739000   0050
 L260,L270,L280,L290,L410,L0$                                                   00740000   0050
SWITCH SWG01:=L140,L150,L160,L170,L180,L190,L200$                               00741000   0050
SWITCH SWG02:=L260,L270,L280$                                                   00742000   0057
COMMENT  SUBROUTINE AVRAGE$                                                     00743000   0062
JNDEVG:=JNOFVG+1$                                                               00744000   0062
JFPART:=JNPART$                                                                 00745000   0064
JINOX:=0$                                                                       00746000   0065
JFGROUP:=JNGROUP$                                                               00747000   0065
JJ:=1$                                                                          00748000   0066
DO BEGIN                                                                        00749000   0067
  SVSTFLUX[JJ]:=0$                                                              00750000   0067
  JI:=1$                                                                        00751000   0068
  DO BEGIN                                                                      00752000   0069
    SVAFLUX[JI,JJ]:=SVAFLUX[JI,JJ]/JFPART$                                      00753000   0069
    SVSAFLUX[JI,JJ]:=SVSAFLUX[JI,JJ]+SVAFLUX[JI,JJ]$                            00754000   0072
    SVSQFLUX[JI,JJ]:=SVSQFLUX[JI,JJ]+SVAFLUX[JI,JJ]**2$                         00755000   0077
    SVSTFLUX[JJ]:=SVSTFLUX[JJ]+SVAFLUX[JI,JJ]$                                  00756000   0082
    END UNTIL (JI:=(JI+1))>JNPCUL$                                              00757000   0085
  SVFFLUX[JJ]:=SVFFLUX[JJ]+SVSTFLUX[JJ]$                                        00758000   0087
  SVDVFLUX[JJ]:=SVDVFLUX[JJ]+SVSTFLUX[JJ]**2$                                   00759000   0089
  END UNTIL (JJ:=(JJ+1))>JNOMAX$                                                00760000   0092
WRITE(PRINT[PAGE])$                                                             00761000   0094
WRITE(PRINT,FL110,LIST1)$                                                       00762000   0098
L115: WRITE(PRINT,FL120)$                                                       00763000   0101
IF JNOMAX>7 THEN GO TO L125$                                                    00764000   0105
JNFORM:=JNOMAX$                                                                 00765000   0106
```

```
GO TO L130;                                                    00760000  0107
L125: JNFORM=7;                                                00767000  0108
L130: GO TO SWGO1(JNFORM);                                     00768000  0108
L140: WRITE(PRINT,FL145);                                      00769000  0111
GO TO L210;                                                    00770000  0114
L150: WRITE(PRINT,FL155);                                      00771000  0115
GO TO L210;                                                    00772000  0118
L160: WRITE(PRINT,FL165);                                      00773000  0119
GO TO L210;                                                    00774000  0122
L170: WRITE(PRINT,FL175);                                      00775000  0123
GO TO L210;                                                    00776000  0126
L180: WRITE(PRINT,FL185);                                      00777000  0127
GO TO L210;                                                    00778000  0130
L190: WRITE(PRINT,FL195);                                      00779000  0131
GO TO L210;                                                    00780000  0134
L200: WRITE(PRINT,FL205);                                      00781000  0135
L210: J1=1;                                                    00782000  0138
DO BEGIN                                                       00783000  0139
  WRITE(PRINT,FL220,LIST2);                                    00784000  0139
  END UNTIL (J1=(J1+1))>JNPCOL;                                00785000  0143
WRITE(PRINT,FL230,LIST1);                                      00786000  0145
IF JNOMAX≤JNFORM THEN GO TO L310;                              00787000  0149
JNFORM=JNOMAX-JNFORM;                                          00788000  0150
WRITE(PRINT(PAGE));                                            00789000  0152
WRITE(PRINT,FL120);                                            00790000  0155
GO TO SWGO2(JNFORM);                                           00791000  0158
L260: WRITE(PRINT,FL265);                                      00792000  0160
GO TO L290;                                                    00793000  0164
L270: WRITE(PRINT,FL275);                                      00794000  0165
GO TO L290;                                                    00795000  0168
L280: WRITE(PRINT,FL285);                                      00796000  0169
L290: J1=1;                                                    00797000  0172
```

```
DO BEGIN                                                                00799000  0173
  WRITE(PRINT,FL220,LIST4);                                             00799000  0173
  END UNTIL (JI+(JI+1))>JNPCOL;                                         00800000  0177
WRITE(PRINT,FL230,LIST5);                                               00801000  0179
L310: WRITE(PRINT,FL320,LIST6);                                         00802000  0183
JJ+1;                                                                   00803000  0187
DO BEGIN                                                                00804000  0188
  JI+1;                                                                 00805000  0188
  DO BEGIN                                                              00806000  0189
    SVAFLUX(JI,JJ)+0;                                                   00807000  0189
    END UNTIL (JI+(JI+1))>JNPCOL END UNTIL (JJ+(JJ+1))>JNOMAX;          00808000  0191
IF JNH1ST<JNHMAX THEN GO TO L0;                                         00809000  0195
IF (XPH+(JINDX))>0 THEN GO TO L0 ELSE IF XPH<0 THEN GO TO L410;         00810000  0197
JINDX+-1;                                                               00811000  0200
JJ+1;                                                                   00812000  0201
DO BEGIN                                                                00813000  0201
  JI+1;                                                                 00814000  0201
  DO BEGIN                                                              00815000  0202
    SVAFLUX(JI,JJ)+SVSAFLUX(JI,JJ)/JFGROUP;                             00816000  0202
    END UNTIL (JI+(JI+1))>JNPCOL;                                       00817000  0206
  SVSTFLUX(JJ)+SVFFLUX(JJ)/JFGROUP;                                     00818000  0206
  END UNTIL (JJ+(JJ+1))>JNOMAX;                                         00819000  0210
WRITE(PRINT[PAGE]);                                                     00820000  0212
WRITE(PRINT,FL400);                                                     00821000  0215
GO TO L115;                                                             00822000  0219
L410: JINDX+1;                                                          00823000  0219
JJ+1;                                                                   00824000  0220
DO BEGIN                                                                00825000  0221
  JI+1;                                                                 00826000  0221
  DO BEGIN                                                              00827000  0222
    SVAFLUX(JI,JJ)+SQRT((SVSQFLUX(JI,JJ)/JFGROUP+2)+(SVSAFLUX(JI,       00828000  0222
     JJ))+2/JFGROUP+3);                                                 00829000  0227
```

```
    END UNTIL (JI>(JI+1))>JNPEOL);                                  00A30000   0230
  SVSTFLUX(JJ1+SQRT((SVDVFLUX[JJ]/JFGROUP**2)-(SVFFLUX[JJ]**2/JFGROUP**3)));  00A31000   0232
  END UNTIL (JJ+(JJ+1))>JNUMAX);                                    00A32000   0236
WRITE(PRINT,PAGE1);                                                 00A33000   0241
WRITE(PRINT,FL460);                                                 00A34000   0244
GO TO L115;                                                         00A35000   0247
L01  BREAK;     END;                                                00A36000   0251
                                                                    0023 IS 0257 LONG, NEXT SEG 0007
PROCEDURE SRANGLE;                                                  00A37000   0314
BEGIN                                                               00A38000   0314
INTEGER  JJ,J1 ;                                                    0314
                                                                    START OF SEGMENT ********** 0025
COMMENT  THE FOLLOWING PROCEDURES ARE USED:  SRRAND;                00A43000   0000
FORMAT FL15(/" NO ANGLE PROBABILITY COULD BE FOUND GREATER THAN",E10.3),  00A44000   0000
                                                                    START OF SEGMENT ********** 0026
FL30(/" INCORRECT SUBSCRIPT FOR ANGLE PROBABILITY,");               00A45000   0000
                                                                    0026 IS 0026 LONG, NEXT SEG 0025
LIST LIST1(JRN);                                                    00A46000   0000
LABEL L50,L20,L35,L45,L40;                                          00A47000   0005
COMMENT  SUBROUTINE ANGLE;                                          00A48000   0005
J1=1;                                                               00A49000   0005
DO BEGIN                                                            00A50000   0006
  SRRANDA(J1,RAS),JRN);                                             00A51000   0006
  JJ=1;                                                             00A52000   0007
  DO BEGIN                                                          00A53000   0008
    IF SVPAGE[JJ]>JRN THEN GO TO L20;                               00A54000   0008
    END UNTIL (JJ+(JJ+1))>JNAG);                                    00A55000   0009
  WRITE(PRINT,FL15,LIST1);                                          00A56000   0011
  JNADA=JNADA+1;                                                    00A57000   0015
  GO TO L50;                                                        00A58000   0016
  L20: IF JJ>1 THEN GO TO L35;                                      00A59000   0017
  WRITE(PRINT,FL30);                                                00A60000   0019
```

```
  JNHDA*JNHDA+1)                                                    00861000   0022
  GO TO L50)                                                        00862000   002A
  L351 SRHANDA(JIBAS&,JHN))                                         00863000   0024
  SVSANG[JI]*SVCANG[JJ-1]*JHN*(SVCANG[JJ-1]*SVCANG[JJ]))            00864000   0026
  IF (XPR*(JNAUP))>0 THEN GO TO LAO ELSE IF XPR<0 THEN GO TO LA5)   00865000   0030
  JPJM1*SVPAG[JJ-1])                                                00866000   0033
  SVWEIGHT[JI]*(1/(SVPAG[JJ]-JPJM1))*(SVCANG[JJ-1]*SVCANG[JJ])/(SVCANG[1   00867000   0035
)*SVCANG[JNAG]))                                                    00868000   0039
  GO TO L50)                                                        00869000   0041
  LA01 SVWEIGHT[JI]*SVHAG[JJ])                                      00870000   0041
  GO TO L50)                                                        00871000   0043
  LA51 SVWEIGHT[JI]*1)                                              00872000   00A4
  L501 END UNTIL (JI*(JI+1))>JNPART)                                00873000   00A5
 END)                                                               00874000   0048
                                                                  0025 IS 0051 LONG, NEXT SEG 0007
PROCEDURE SHPATHL)                                                  00875000   0314
BEGIN                                                               00876000   0314
  INTEGER JJ)      REAL ADJUST )                                                 0314
                                                                  START OF SEGMENT ********** 0027
COMMENT  THE FOLLOWING PROCEDURES ARE USED)  SRRANDA)                00881000   0000
FORMAT FL130(/" LOC =",I4,"  J =",I4,"  JHR =",I4,"  JHT =",I4,"  KN =",   00882000   0000
                                                                  START OF SEGMENT ********** 0028
S1,E10.3/"  HHO =",S1,E10.3,"  COTH =",S1,E10.3,"  TAUH1 =",S1,E10.3,   00883000   0000
"  TAUH2 =",S1,F10.3/"  PL =",S1,F10.3,"  H2 =",S1,E10.3))              00884000   0000
                                                                  0028 IS 0041 LONG, NEXT SEG 0027
LIST LIST1(JLOC,JJ,JJHR,JJH1,JHN,JHHO,JCOTH,JTAUH1,JTAUH2,JPL,JH2))  00885000   0000
LABEL L20,L30,L50,L58,L105,L70,L100,L110,L0)                        00886000   0018
SRHANDA(JIBAS2,JHN))                                                00887000   0018
JLOC*25)                                                            00888000   0019
JPL*0)                                                              00889000   0020
IF ABS(JCOTH)SJSMVAL THEN GO TO L20)                                00890000   0021
IF JCOTH>0 THEN GO TO L30)                                          00891000   0022
```

```
L20: JRHO*=LN(JRN);                                                    00A92000  0023
GO TO L50;                                                             00A93000  0025
L30: JUPLMIT*(SVTAU(JNOH)-JTAUM1)/JCOTH;                               00A94000  0026
JAOJUST*1-EXP(-JUPLM(T));                                              00A95000  0030
JRHO*=LN(1-JRN*JAOJUST);                                               00A96000  0032
JMAIT*JMAIT*JAOJUST;                                                   00A97000  0035
L50: JTAUM2*JTAUM1+JRHO*JCOTH;                                         00A98000  0036
IF JTAUM2>0 THEN GO TO L58;                                            00A99000  0038
JTAUM2*0;                                                              00900000  0040
JJMB*1;                                                                00901000  0040
JJMT*2;                                                                00902000  0041
JM2*=JOLONG;                                                           00903000  0042
GO TO L105;                                                            00904000  0043
L58: JJ*1;                                                             00905000  0044
DO BEGIN                                                               00906000  0046
  IF JTAUM2<SVTAU(JJ) THEN GO TO L70;                                  00907000  0046
  END UNTIL (JJ*(JJ+1))>JNUM;                                          00908000  0048
JJMB*JNUM-1;                                                           00909000  0050
JJMT*JNUM;                                                             00910000  0051
JM2*JOLONG;                                                            00911000  0052
GO TO L105;                                                            00912000  0053
L70: JJMB*JJ-1;                                                        00913000  0053
JJMT*JJ;                                                               00914000  0055
IF ABS(JCOTH)>JSMVAL THEN GO TO L100;                                  00915000  0056
JM2*JM1;                                                               00916000  0057
JPL*JRHO/((SVTAU(JJMT)-SVTAU(JJMB))/(SVMV(JJMT)-SVMV(JJMB)));          00917000  0058
GO TO L110;                                                            00918000  0062
L100: JM2*SVMV(JJMB)+(SVMV(JJMT)-SVMV(JJMB))*(JTAUM2-SVTAU(JJMB))/(SVTAU  00919000  0062
(JJMT)-SVTAU(JJMB));                                                   00920000  0066
L105: JPL*(JM2-JM1)/JCOTH;                                             00921000  0068
L110: IF JTDUMP>0 THEN GO TO L0;                                       00922000  0070
WRITE(PRINT,FL130,LIST);                                               00923000  0072
```

```
LOI  END;                                          00934000    0076
                                                 0027 IS 0079 LONG, NEXT SEG 0007
PROCEDURE SRINITAL;                                00935000    0314
BEGIN                                              00936000    0314
INTEGER JJ,JI,JK,JN ;                                          0314
                                                 START OF SEGMENT ********** 0029
COMMENT  SUBROUTINE INITAL;                        00934000    0000
JJ=1;                                              00935000    0000
DO BEGIN                                           00936000    0000
  JLR=JNPCOL+1;                                    00937000    0000
  J1=1;                                            00938000    0002
  DO BEGIN                                         00939000    0002
    SVSAFLUX[J1,JJ]=0;                             00940000    0002
    SVSWFLUX[J1,JJ]=0;                             00941000    0004
    END UNTIL (JI=(JI+1))>JLR;                     00942000    0006
  JK=1;                                            00943000    0006
  DO BEGIN                                         00944000    0009
    SVRUD[JK,JJ]=0 END UNTIL (JK=(JK+1))>JMAXR;    00945000    0009
  JN=1;                                            00946000    0014
  DO BEGIN                                         00947000    0014
    SVFLUD[JN,JJ]=0;                               00948000    0014
    END UNTIL (JN=(JN+1))>JNRMAX;                  00949000    0016
  SVRFLUX[JJ]=0;                                   00950000    0019
  SVFFLUX[JJ]=0;                                   00951000    0029
  SVUVFLUX[JJ]=0;                                  00952000    0021
  SVFLUN[JJ]=0;                                    00953000    0022
  END UNTIL (JJ=(JJ+1))>JNUMAX;                    00954000    0024
JMAXR1=JMAXR+1;                                    00955000    0026
JJAMAX=JNUMAX*JNAZA;                               00956000    0027
JJ=1;                                              00957000    0026
DO BEGIN                                           00958000    002V
  JI=1;                                            00959000    002V
```

```
  DO BEGIN                                                          00960000    0030
    JK=1;                                                           00961000    0030
    DO BEGIN                                                        00962000    0031
      SVTFLUX[JJ,JI]=0;                                             00963000    0031
      SVFLUX[JK,JJ,JI]=0 END UNTIL (JK=(JK+1))>JNPA END UNTIL (JI=( 00964000    0033
   JI+1))>JJAMAX END UNTIL (JJ=(JJ+1))>JMAXM1;                      00965000    0038
JMAXCOL=INT(0);                                                     00966000    0042
JNMAJ=INT(0);                                                       00967000    0043
JNMSTOP=0;                                                          00968000    00A5
JNMAXM=0;                                                           00969000    00A5
JI=1;                                                               00970000    00A6
DO BEGIN                                                            00971000    00A7
  SVNMI[0,JI]=INT(0) END UNTIL (JI=(JI+1))>JNRMAX;                  00972000    00A7
 END;                                                               00973000    0051
                                                                  0029 IS 035A LONG, NEXT SEG 0007
PROCEDURE SHREFLCT;                                                 0097A000    0314
BEGIN                                                               00975000    031A
REAL  JDFNDM;       INTEGER  JI,JJAIL;                                          031A
                                                                  START OF SEGMENT ********** 0030
COMMENT  THE FOLLOWING PROCEDURES ARE USED:  SRRANDA;               00981000    0000
FORMAT FL35(/" REFLECTION ANGLE DISTRIBUTION FOR BOUNDARY",I3,      00982000    0000
                                                                  START OF SEGMENT ********** 0031
 " IS IN ERROR.");                                                  00983000    0000
                                                                  0031 IS 0016 LONG, NEXT SEG 0030
LIST LIST1(JNRM);                                                   0098A000    0000
LABEL L10,L20,L15,L70,L50,LA0,LA0,L0;                               00985000    0005
SWITCH SWG0I=L10,L20,L15,L70;                                       0098A000    0005
COMMENT  SUBROUTINE REFLCT;                                         00987000    0011
SRRANDA(JIHASS,JRN);                                                00988000    0011
JJAIL=SVJREFLTI[JNRR];                                              00989000    0013
GO TO SWG0I[JJAIL];                                                 00990000    0014
L10: JCOTHI=JRN;                                                    00991000    0016
```

```
GO TO L70;                                                          00992000   0016
L15: JCOTH1=-JRN;                                                   00993000   0017
GO TO L70;                                                          00994000   0019
L20: JFNRA=SVNRFCOS(JNRB);                                          00995000   0019
JPRI=JRN*JFNRA;                                                     00996000   0021
JI=INT(JPRI);                                                       00997000   0022
IF (XPR=(JI))>0 THEN GO TO L60 ELSE IF XPR=0 THEN GO TO L50;        00998000   0023
WRITE(PRINT,FL35,LIST1);                                            00999000   0026
JNRDA=JNRDA+1;                                                      01000000   0030
GO TO L01;                                                          01001000   0031
L50: IF (XPR=(JJA1(L-2)))=0 THEN JCOTH1=1=JPRI*(SVRFLCOS(I,JNRB)=1) ELSE    01002000   0032
  JCOTH1=JPRI*SVRFLCOS(1,JNRB);                                     01002100   0037
GO TO L70;                                                          01003000   0040
L60: JFI=JI;                                                        01004000   0040
JCOTH1=SVRFLCOS(JI,JNRB)=(JPRI-JFI)*(SVRFLCOS(JI+1,JNRB)-SVRFLCOS(  01005000   0041
  JI,JNRB));                                                        01006000   0045
L70: JSITH1=SQRT(1-JCOTH1=2);                                       01007000   0045
L80: SRHANDA(JIRASE,JRN);                                           01008000   0050
JSPT=2*JRN-1;                                                       01009000   0052
SRHANDA(JIRAS1,JRN);                                                01010000   0053
JCPT=2*JRN-1;                                                       01011000   0054
JDENOM=JCPT=2+JSPT=2;                                               01012000   0056
IF JDENOM>1 THEN GO TO L80;                                         01013000   0058
JDENOM=SQRT(JDENOM);                                                01014000   0060
JCPHI1=JCPT/JDENOM;                                                 01015000   0061
JSPHI1=JSPT/JDENOM;                                                 01016000   0062
JC4PHI=JCPHI1*JCDAZI-JSPHI1*JSU4ZI;                                 01017000   0064
JS4PHI=JSPHI1*JCD4ZI+JCPHI1*JSU4ZI;                                 01018000   0066
L01:  END;                                                          01019000   0068
                                                    0030 IS 0074 LONG, NEXT SEG 0007

PROCEDURE SHSCTANG;                                                 01020000   0314
BEGIN                                                               01021000   0314
```

```
REAL  JCOPHI, JSOPHI)  INTEGER  J1,JNPASE )                                  0314
                                                    START OF SEGMENT ********** 0032
COMMENT  THE FOLLOWING PROCEDURES ARE USED:  SRREFLCT,SRRANDA)      01029000  0000
FORMAT FL80(/" THE PHASE ANGLE PROBABILITIES FOR MATERIAL",I3,      01030000  0000
                                                    START OF SEGMENT ********** 0033
 " ARE INCORRECT,"),                                01031000  0000
FL130(/" LOC =",I4," NPHASE =",I4," NCM =",I4," REFL =",S1,E10.3,   01032000  0000
 "  CSANG =",S1,F10.3/" SSANG =",S1,F10.3," CTEP =",S1,E10.3,      01033000  0000
 "  STEP =",S1,E10.3," DEG =",S1,F10.3," COPHI =",S1,E10.3/        01034000  0000
 " SOPHI =",S1,E10.3," COTH2 =",S1,F10.3," SITH2 =",S1,E10.3,      01035000  0000
 "  SOEPHI =",S1,E10.3/" COEPH1 =",S1,E10.3," CPHI2 =",S1,E10.3,   01036000  0000
 "  SPHI2 =",S1,E10.3," COTH1 =",S1,E10.3/" SITH1 =",S1,E10.3,     01037000  0000
 "  CPHI1 =",S1,F10.3," SPHI1 =",S1,F10.3," RN =",S1,E10.3,        01038000  0000
 "  CAPHI =",S1,F10.3," SAPHI =",S1,F10.3))                                  0000
                                                    0033 IS 0115 LONG, NEXT SEG 0032
LIST LIST1(JNCM))                                   01040000  0000
LIST LIST2(JLOC,JNPHASE,JNCM,JREFL,JCSANG,JSSANG,JCTEP,JSTEP,JDEDM,  01041000  0005
 JCOPHI,JSOPHI,JCOTH2,JSITH2,JSOEPHI,JCOEPHI,JCPHI2,JSPHI2,JCOTH1,  01042000  0017
 JSITH1,JCPHI1,JSPHI1,JRN,JCAPHI,JSAPHI))                           01043000  0029
LABEL L5,L137,L10,L50,L120,L100,L110,L130,L136,L0,L150)              01044000  0039
COMMENT  SUBROUTINE SCTANG)                         01045000  0039
IF JREFL<0 THEN GO TO L5)                           01046000  0039
SRREFLCT)                                           01047000  0040
GO TO L137)                                         01048000  0041
L5: SRRANDA(J1RAS3,JRN))                            01049000  0041
IF JRN>JRATLEE THEN GO TO L50)                      01050000  0043
L10: SRRANDA(J1RAS4,JRN))                           01051000  0044
JCSANG=1-2*JRN)                                     01052000  0046
SRRANDA(J1RAS5,JRN))                                01053000  0047
IF JRN<.5 THEN GO TO L120)                          01054000  0048
SRRANDA(J1RASE,JRN))                                01055000  0050
IF JRN<JCSANG*JCSANG THEN GO TO L120 ELSE GO TO L10)  01056000  0051
```

```
L50: SRNANDA(J1NAS1,JRN);                                    01057000  0053
JFNPA+SVNPHANG(JNCM);                                        01058000  0055
JPR1+JRN*JFNPA;                                              01059000  0056
JI+INT(JPRI);                                                01060000  0057
IF (XPR+(JI))>0 THEN GO TO L110 ELSE IF XPN=0 THEN GO TO L100; 0106100C  0058
WRITE(PNIN1,FL80,LIST1);                                     01062000  0061
JWHOA+JWHOA+1;                                               01063000  0065
GO TO L0;                                                    0106A000  0066
L100: JCSANG+1+JPR1*(SVPHANG(1,JNCM)-1);                     01065000  0068
GO TO L120;                                                  0106A000  0071
L110: JFI+JI;                                                01067000  0071
JCSANG+SVPHANG(JI,JNCM)+(JPR1-JFI)*(SVPHANG(JI+1,JNCM1-SVPHANG(JI,  01068000  0072
 JNCM));                                                     01069000  0077
L120: JSSANG+SQRT(1-JCSANG*JCSANG);                          01070000  0079
L130: SRNANDA(J1RAS2,JRN);                                   01071000  0081
JCTEP+1-2*JRN;                                               01072000  0083
SNMANDA(J1RAS3,JNN);                                         01073000  0084
JSTEP+1-2*JRN;                                               0107A000  0085
JDEDM+JCTEP+2+JSTEP+2;                                       01075000  0087
IF JDEDM>1 THEN GO TO L130;                                  0107A000  0089
JDEDM+SQRT(JDEDM);                                           01077000  0091
JCDPM1+JCTEP/JDEDM;                                          0107A000  0092
JSDPM1+JSTEP/JDEDM;                                          01079000  0093
IF JS1TH2>JSMALL THEN GO TO L136;                            01080000  0095
JCOTH1+JCSANG*JCOTH2;                                        01081000  0096
JS1TH1+JSSANG;                                               01082000  0097
JCPM11+JCDPM1;                                               01083000  0098
JSPM11+JSDPM1;                                               01084000  0099
    JCDEPM1+JCDPM1;                                          01084100  0099
    JSDEPM1+JSDPN1;                                          01084200  0100
    GO TO L150;                                              01085000  0101
L136: JCOTH1+JCOTH2*JCSANG+JS1TH2*JSSANG*JCDPM1;             0108A000  010A
```

```
JS1TH1=SQRT(1-JC0TH1*JC0TH1);                                  01087000   0106
JSDEPH1=(J3SANG*JSDPH1)/JS1TH1;                                01088000   0109
JC0EPH1=(JCSANG-JC0TH2*JC0TH1)/(JS1TH2*JS1TH1);                01089000   0111
JCPH11=JCPH12*JC0EPH1-JSPH12*JSDEPH1;                          01090000   0113
JSPH11=JSPH12*JC0EPH1+JCPH12*JSDEPH1;                          01091000   0116
    L150:                                                      01091500   0118
JCAPH11=JCAPH1;                                                01092000   0119
JSAPH11=JSAPH1;                                                01093000   0119
JCAPH1=JCAPH11*JC0EPH1-JSAPH11*JSDEPH1;                        01094000   0120
JSAPH1=JSAPH11*JC0EPH1+JCAPH11*JSDEPH1;                        01095000   0122
L137: JC0TH2=JC0TH1;                                           01096000   0125
JS1TH2=JS1TH1;                                                 01097000   0125
JCPH12=JCPH11;                                                 01098000   0126
JSPH12=JSPH11;                                                 01099000   0127
JL0C=80;                                                       01100000   0128
IF J OUMPS0 THEN GO TO L01;                                    01101000   0128
WRITE(PRINT,FL139,LIST2);                                      01102000   0130
L01:  END;                                                     01103000   0133
                                                    0032 IS 0134 LONG, NEXT SEG 0007
PROCEDURE SROMEAM;                                             01104000   031A
BEGIN                                                          01105000   031A
INTEGER JJ, JJ2;  REAL   JVH;                                             031A
                                                    START OF SEGMENT ********** 003A
FORMAT FL11(" HS IS GREATER THAN HV(NUM),                ");   01110000   0000
                                                    START OF SEGMENT ********** 0035
FL230(" RADIATION RESEARCH ASSOCIATES -LITE- PROBLEM",110),    01111000   0000
FL240(/" DIRECT BEAM LIGHT INTENSITIES"//                      01112000   0000
 "  DETECTOR   DIRECT INTENSITY"),                             01113000   0000
FL250(/" ",16,X8,S1,E11.3);                                               0000
                                                    0035 IS 00AY LONG, NEXT SEG 003A
LIST (LIST1(JNPHOR));                                          01115000   0000
LIST (LIST2(JJ,SVDBFLUX(JJ)));                                 01116000   0005
```

```
LABEL L3,L100,L210,L0;
COMMENT  SUBROUTINE OBEAM;
JJ2:=2;
DO BEGIN
  IF JHS≤SVHV[JJ2] THEN GO TO L3;
  END UNTIL [JJ2:=[JJ2+1]]>JNUM;
WRITE[PRINT,FL11];
GO TO L0;
L3: JJHB:=JJ2-1;
JJHT:=JJ2;
JJ:=1;
DO BEGIN
  JVD:=SVHD[JJ]-JHS;
  JT:=SQRT(JVD↑2+SVRD[JJ]↑2);
  JCOTH:=JVD/JT;
  IF ABS[JCOTH]>JSMV&L THEN GO TO L100;
  JRHOT:=JT×(SVTAU[JJHT]-SVTAU[JJHB])/(SVHV[JJHT]-SVHV[JJHB]);
  GO TO L210;
  L100: JRHOT:=[SVTAUHD[JJ]-JTAUH)/JCOTH;
  L210: SVDBFLUX[JJ]:=SVDBS0[JJ]×EXP(-JRHOT)/JT↑2;
  END UNTIL (JJ:=(JJ+1))>JNUM&X;
WRITE(PRINT[PAGE]);
WRITE(PRINT,FL230,LIST);
WRITE(PRINT,FL240);
JJ:=1;
DO BEGIN
  WRITE(PRINT,FL250,LIST2);
  END UNTIL (JJ:=(JJ+1))>JNUM&X;
WRITE(PRINT[PAGE]);
JHHDA:=JHHDA+1;
L0:  END;
```

```
01117000   0012
01118000   0012
01119000   0012
01120000   0013
01121000   0013
01122000   0014
01123000   0017
01124000   0020
01125000   0021
01126000   0022
01127000   0023
01128000   0023
01129000   0023
01130000   0025
01131000   0028
01132000   0029
01133000   0031
01134000   0035
01135000   0037
01136000   0039
01137000   0043
01138000   0045
01139000   0048
01140000   0052
01141000   0055
01142000   0056
01143000   0056
01144000   0060
01145000   0062
01146000   0065
01147000   0067
0034 IS 0071 LONG, NEXT SEG 0007
```

```
PROCEDURE SNCHECK;                                              01158000   0314
BEGIN                                                           01159000   031A
INTEGER   JJ,JINAG,JINPA,JINPCOL,JINRF1,JINRF2,JINRF,JJCHECH,JJCHECK;    952000   031A
                                                                START OF SEGMENT ********** 0036
        JJ,JNRF1,JNRF2,JNRF3,JNNF,JNAG1,JNPA1,JNPCOL1 ;         953000   0000
FORMAT FL25(" THE NUMBER OF REFLECTION BOUNDRIES",I3,            01158000   0000
                                                                START OF SEGMENT ********** 0037
 " EXCEEDS THE LIMIT OF 5 ALLOWED","DATA CHECK CONTINUES..."),   01159000   0000
FL45(" THE NUMBER OF DETECTIONS",I3," EXCEEDS THE LIMIT OF 10 ALLOWED",  01160000   0000
 "DATA CHECK CONTINUES..."),                                     01161000   0000
FL65(" THE NUMBER OF MATERIALS",I3," EXCEEDS THE LIMIT OF 10 ALLOWED",   01162000   0000
 "DATA CHECK CONTINUES..."),                                     01163000   0000
FL85(" THE NUMBER OF PRINT COLLISIONS",I3,                       01164000   0000
 " EXCEEDS THE LIMIT OF 24 ALLOWED","DATA CHECK CONTINUES..."),  01165000   0000
FL105(" THE NUMBER OF PRINT ANGLES",I3,                          01166000   0000
 " EXCEEDS THE LIMIT OF 25 ALLOWED","DATA CHECK CONTINUES..."),  01167000   0000
FL125(" THE NUMBER OF SOURCE ANGLES",I3,                         01168000   0000
 " EXCEEDS THE LIMIT OF 37 ALLOWED","DATA CHECK CONTINUES..."),  01169000   0000
FL145(" THE NUMBER OF REGIONS",I4," EXCEEDS THE LIMIT OF 100 ALLOWED",   01170000   0000
 "DATA CHECK CONTINUES..."),                                     01171000   0000
FL165(" THE NUMBER OF BOUNDRIES",I4,                             01172000   0000
 " EXCEEDS THE LIMIT OF 100 ALLOWED","DATA CHECK CONTINUES..."), 01173000   0000
FL180(" COSINE SOURCE ANGLES MUST BE INPUT IN DESCENDING ORDER",         01174000   0000
 "DATA CHECK CONTINUES..."),                                     01175000   0000
FL215(" COSINE PRINT ANGLES MUST BE INPUT IN DESCENDING ORDER",  01176000   0000
 "DATA CHECK CONTINUES..."),                                     01177000   0000
FL235(" REFLECTION ANGLES MUST BE INPUT IN DESCENDING ORDER",    01178000   0000
 "DATA CHECK CONTINUES..."),                                     01179000   0000
FL270(" REFLECTION COSINES MUST BE INPUT IN DESCENDING ORDER",   01180000   0000
 "DATA CHECK CONTINUES..."),                                     01181000   0000
FL315(" DIFFERENTIAL COSINES MUST BE INPUT IN DESCENDING ORDER",  01182000   0000
 "DATA CHECK CONTINUES...");                                     01183000   0000
```

```
FL355(*  PHASE ANGLES MUST BE INPUT IN OFSCENDING ORDER*,                    01184000    0000
 *,DATA CHECK CUNTINUES,..*),                                               01185000    0000
FL385(*  ANGLE PROBABILITIES MUST BE INPUT IN ASCENDING ORDER*,              01186000    0000
 *,DATA CHECK CUNTINUES,..*),                                               01187000    0000
                                                                  0037 IS 0267 LONG, NEXT SEG 0038
                                                                  START OF SEGMENT ********** 0038
FL415(*  INPUT NUMBER OF COLLISSION MUST BF IN ASCENDING ORDEN*,             01188000    0000
 *,DATA CHFCK CONTINUES,..*),                                               01189000    0000
FL435(* *,*  THERE ARE A TOTAL OF*,I3,* INPUT DATA ERRORS*///               01190000    0000
 * AKE PRORLEM OFF COMPUTEN ANU CORRECT ERRORS. BETTER LUCK NEXT *,         01191000    0000
 *TIME*),                                                                   01192000    0000
FL455(* INPUT UATA SEEMS TU BE ALLRIGHT. EXECUTION CONTINUES.*);            01193000    0000
                                                                  0038 IS 0056 LONG, NEXT SEG 0036
LIST LIST1(JNRFLR);                                                         01194000    0000
LIST LIST2(JNOMAX);                                                         01195000    0005
LIST LIST3(JNMAT);                                                          01196000    0010
LIST LIST4(JNPCUL);                                                         01197000    0015
LIST LIST5(JNPA);                                                           01198000    0020
LIST LIST6(JNAG);                                                           01199000    0025
LIST LIST7(JNRMAX);                                                         01200000    0030
LIST LIST8(JNBMAX);                                                         01201000    0035
LIST LIST9(JJCHECK);                                                        01202000    0040
LABEL L30,L50,L70,L90,L110,L130,L150,L170,L200,L220,L300,L240,L280,          01203000    0045
 L370,L320,L360,L390,L420,L450;                                             01204000    0045
JJCHECK=0;                                                                  01205000    0045
IF JNRFLR55 THEN GO TO L30;                                                 01206000    0046
WRITE(PRINT,FL25,LIST1);                                                    01207000    0047
JJCHECK=JJCHECK+1;                                                          01208000    0051
L30: IF JNOMAX$10 THEN GO TO L50;                                           01209000    0052
WRITE(PRINT,FL45,LIST2);                                                    01210000    0054
JJCHECK=JJCHECK+1;                                                          01211000    0056
L50: IF JNMAT$10 THEN GO TO L70;                                            01212000    0059
```

```
WRITE(PRINT,FL65,L1ST3);                                    01213000  0061
JJCHECK=JJCHECK+1;                                          01214000  0065
L70: IF JNPCOLS24 THEN GO TO L90;                           01215000  0066
WRITE(PRINT,FL85,L1ST4);                                    01216000  0068
JJCHECK=JJCHECK+1;                                          01217000  0072
L90: IF JNPAS25 THEN GO TO L110;                            01218000  0073
WRITE(PRINT,FL105,L1ST5);                                   01219000  0075
JJCHECK=JJCHECK+1;                                          01220000  0079
L110: IF JNAGS37 THEN GO TO L130;                           01221000  0080
WRITE(PRINT,FL125,L1STA);                                   01222000  0082
JJCHECK=JJCHECK+1;                                          01223000  0086
L130: IF JNRMAXS100 THEN GO TO L150;                        01224000  0087
WRITE(PRINT,FL145,L1ST7);                                   01225000  0089
JJCHECK=JJCHECK+1;                                          01226000  0093
L150: IF JNRMAXS100 THEN GO TO L170;                        01227000  0094
WRITE(PRINT,FL165,L1STA);                                   01228000  0096
JJCHECK=JJCHECK+1;                                          01229000  0100
J1NAG=JNAG-1;                                               01230000  0101
L170: JJ=1;                                                 01231000  0102
DO BEGIN                                                    01232000  0103
  IF SVCANG1[JJ]=SVCANG[JJ+1] THEN GO TO L200;              01233000  0103
  WRITE(PRINT,FL180);                                       01234000  0106
  JJCHECK=JJCHECK+1;                                        01235000  0109
  L200: END UNTIL (JJ=(JJ+1))>J1NAG;                        01236000  0110
J1NPA=JNPA-1;                                               01237000  0113
JJ=1;                                                       01238000  0114
DO BEGIN                                                    01239000  0115
  IF SVC1PA[JJ]=SVC1PA[JJ+1] THEN GO TO L220;               01240000  0115
  WRITE(PRINT,FL215);                                       01241000  0117
  JJCHECK=JJCHECK+1;                                        01242000  0121
  L220: END UNTIL (JJ=(JJ+1))>J1NPA;                        01243000  0122
IF JNRFLAS0 THEN GO TO L300;                                01244000  0125
```

```
JI1+1;                                                          01245000  0126
DO BEGIN                                                        01246000  0127
  JNRF+SVNRFANG[JI1];                                           01247000  0127
  JINRF+JNRF-1;                                                 01248000  0128
  JJ+1;                                                         01249000  0129
  DO BEGIN                                                      01250000  0130
    IF SVRFANG[JJ,JI11≥SVRFANG[JJ+1,JI11 THEN GO TO L240;       01251000  0130
    WRITE(PRINT,FL235);                                         01252000  0134
    JJCHECK+JJCHECK+1;                                          01253000  0137
    L240: END UNTIL (JJ+(JJ+1)-1>JINRF;                         01254000  0138
  END UNTIL (JI1+(JI1+1)-1>JNRFLR;                              01255000  0141
JI1+1;                                                          01256000  0143
DO BEGIN                                                        01257000  0144
  JNRF1+SVNRFCUS[JI1];                                          01258000  0144
  JINRF1+JNRF1-1;                                               01259000  0145
  JJ+1;                                                         01260000  0146
  DO BEGIN                                                      01261000  0147
    IF SVRFLCUS[JJ,JI1]≥SVRFLCUS[JJ+1,JI1] THEN GO TO L280;     01262000  0147
    WRITE(PRINT,FL270);                                         01263000  0151
    JJCHECK+JJCHECK+1;                                          01264000  0154
    L280: END UNTIL (JJ+(JJ+11)>JINRF1;                         01265000  0155
  END UNTIL (JI1+(JI1+11)>JNRFLR;                               01266000  0158
L300: JI1+1;                                                    01267000  0160
DO BEGIN                                                        01268000  0161
  IF SVRAYLEE[JI11=1 THEN GO TO L370;                           01269000  0161
  JNRF2+SVNOFCUS[JI1];                                          01270000  0163
  JINRF2+JNRF2-1;                                               01271000  0164
  JJ+1;                                                         01272000  0165
  DO BEGIN                                                      01273000  0166
    IF SVOIFCOS[JJ,JI11≥SVOIFCOS[JJ+1,JI11 THEN GO TO L320;     01274000  0166
    WRITE(PRINT,FL315);                                         01275000  0170
    JJCHECK+JJCHECK+1;                                          01276000  0173
```

```
    L320; END UNTIL (JJ+(JJ+1))>JINRF2;                    01277000   0174
  JNRF3+SVNPHANG1J11]-1;                                   01278000   0177
  JJ+1;                                                    01279000   0178
  DO BEGIN                                                 01280000   0179
    IF SVPHANG1JJ,J11]≥SVPHANG1JJ+1,J11] THEN GO TO L360;  01281000   0179
    WRITE(PRINT,FL355);                                    01282000   0183
    JJCHECK+JJCHECK+1;                                     01283000   0186
    L360; END UNTIL (JJ+(JJ+1))>JNRF3;                     01284000   0188
  L370; END UNTIL (J11+(J11+1))>JNMAT;                     01285000   0190
JJ+1;                                                      01286000   0193
DO BEGIN                                                   01287000   0194
  IF SVPAG[JJ]≤SVPAG1JJ+1] THEN GO TO L390;                01288000   0194
  WRITE(PRINT,FL385);                                      01289000   0196
  JJCHECK+JJCHECK+1;                                       01290000   0199
  L390; END UNTIL (JJ+(JJ+1))>JINAG;                       01291000   0201
JINPCOL+JNPCOL-1;                                          01292000   0203
JJ+1;                                                      01293000   0204
DO BEGIN                                                   01294000   0205
  IF SVINCOL1JJ]≤SVINCOL1JJ+1] THEN GO TO L420;            01295000   0205
  WRITE(PRINT,FL415);                                      01296000   0207
  JJCHECK+JJCHECK+1;                                       01297000   0211
  L420; END UNTIL (JJ+(JJ+1))>JINPCOL;                     01298000   0212
IF JJCHECK≤0 THEN GO TO L450;                              01299000   0215
WRITE(PRINT(PAGE));                                        01300000   0216
WRITE(PRINT,FL435,LIST9);                                  01301000   0219
ERROR(0);                                                  01302000   0223
L450; WRITE(PRINT,FL455);                                  01303000   0224
 END;                                                      01304000   0228
                                                  0036 IS 0237 LONG, NEXT SEG 0007
PROCEDURE SRMAIN;                                          01305000   0314
BEGIN                                                      01306000   0314
INTEGER JJ2, JJ#1L;                                                   0314
```

```
                                                                    START OF SEGMENT ********** 0039
REAL JCRATIO, JFRACT)                                                                          0000
COMMENT  THE FOLLOWING PROCEDURES ARE USED:  SRINITAL,SHSEARCH,SRAVRAGE,        01321000    0000
SRANGLE,SHPATHL,SRDSTBD,SRMANOA,SRDETECT,SHSCTANG)                              01322000    0000
FORMAT FL11(" HS IS GREATER THAN HV(NDH).                      "),              01323000    0000
                                                                    START OF SEGMENT ********** 0040
FL86(/" CANNOT LOCATE REGION CONTAINING SOURCE PARTICLE."),                     01324000    0000
FL76(/" LOC =",I4," NPART =",I4," NSP =",I4," NHIST =",I6," NCH =",             01325000    0000
 I4," NCOL =",I4/" H1 =",S1,E10.3," R1 =",S1,E10.3," COTH1 =",S1,               01326000    0000
 E10.3," SITH1 =",S1,E10.3/" CPHI1 =",S1,E10.3," SPHI1 =",S1,E10.3,             01327000    0000
 " WAIT =",S1,E10.3),                                                                      0000
FL96(/" LOC =",I4," NCR =",I4," NCH =",I3," R =",S1,E10.3," H =",S1,            01329000    0000
E10.3/" COTH =",S1,E10.3," SITH =",S1,E10.3," CIPH =",S1,E10.3,                 01330000    0000
 " SPH1 =",S1,E10.3),                                                                      0000
FL106(/" A NEGATIVE OR ZERO PATH LENGTH WAS GENERATED, PL=",S1,E10.3),          01332000    0000
FL136(/" PROGRAM FAILED TO CALCULATE DISTANCE TO A BOUNDARY."),                 01333000    0000
FL142(/" LOC =",I4," NCR =",I4," NCH =",I4," T =",S1,E10.3,                     01334000    0000
 " SUMOST =",S1,E10.3/" DIST =",S1,E10.3," RHOT =",S1,E10.3," DT =",            01335000    0000
 S1,E10.3," HT =",S1,E10.3/" RHO =",S1,E10.3," NCH =",I4," NLH =",I4)           01336000    0000
,                                                                               01337000    0000
FL147(/" LOC =",I4," NCH =",I4," NLH =",I4," H =",S1,E10.3," TS =",             01338000    0000
 S1,E10.3/" RT =",S1,E10.3," CPHI =",S1,E10.3," R =",S1,E10.3),                 01339000    0000
FL177(/" CANNOT FIND REGION CONTAINING PARTICLE COORDINATES, H=",S1,            01340000    0000
 E10.3," R=",S1,E10.3),                                                                    0000
FL26A(/" LOC =",I4," NCR1 =",I4," NCR2 =",I4," DIST =",S1,E10.3,                01342000    0000
 " DT =",S1,E10.3/" T =",S1,E10.3," SUMOST =",S1,E10.3," H2 =",S1,              013A3000    0000
 E10.3," TS =",S1,E10.3/" HT =",S1,E10.3," CPHI2 =",S1,E10.3," R2 =",           013A4000    0000
S1,E10.3," SPH12 =",S1,E10.3/" COTH2 =",S1,E10.3," SITH2 =",S1,E10.3,          013A5000    0000
 " NCOL =",I6))                                                                 013A6000    0000
                                                                    0040 IS 0289 LONG, NEXT SEG 0039
LIST LIST1(JLOC,JNPART,JNSP,JNHIST,JNCR,JNCOL,JH1,JR1,JCOTH1,JSITH1,            013A7000    0000
 JCPHI1,JSPH11,JWAIT))                                                          01348000    0014
```

```
LIST LIST2(JLOC,JNCR,JNCH,JH,JH,JCOTH,JSITH,JCPH1,JSPH1));                    01349000    0020
LIST LIST3(JPL));                                                             01350000    0035
LIST LIST4(JLOC,JNCR,JNCR,JI,JSUMPST,JOIST,JRHOT,JOT,JHT,JRHO,JNCH,           01351000    0040
 JNLM));                                                                      01352000    0055
LIST LIST5(JLOC,JNCH,JNLM,JH,JTS,JRT,JCPH1,JR));                              01353000    0059
LIST LIST6(JH,JH));                                                           01354000    0073
LIST LIST7(JLOC,JNCR1,JNCH2,JOIST,JRT,JT,JSUMOST,JH2,JTS,JRT,JCPH12,          01355000    0080
 JH2,JSPH12,JCOTH2,JSITH2,JNCOL));                                            01356000    0095
BEGIN                                                                         01357000    0104
LABEL L3,L2,L8,L380,L7,L10,L60,L70,L80,L100,L110,L130,L140,L144,L250,          01358000    0104
                                                                              OF SEGMENT ********** 0041
 L550,L600,L150,L170,L1400,L165,L161,L166,L260,L180,L188,L310,L257,           01359000    0000
 L258,L268,L320,L0,L1800;                                                     01360000    0000
SWITCH SWG0:=L165,L165,L165,L151,L161;                                        01361000    0000
COMMENT  SUBROUTINE MAIN(PLANF));                                             01362000    0005
JNPART=JNHMAX OIV JNGROUP;                                                    01363000    0005
JNSP=JNPART+1;                                                                01364000    0007
JNH1ST=0;                                                                     01365000    0008
JNUEVG=0;                                                                     01366000    0009
SWINITAL;                                                                     01367000    0010
JMPREG=JNSOREG;                                                               01368000    0010
JHH0A=0;                                                                      01369000    0011
JH=JHS;                                                                       01370000    0012
JH=0;                                                                         01371000    0012
JJ2=2;                                                                        01372000    0013
DO BEGIN                                                                      01373000    0014
  IF (XPR=(JHS-SVHV[JJ21))<0 THEN GO TO L3 ELSF IF XPR=0 THEN GO TO           01374000    0014
   L2;                                                                        01375000    0017
  END UNTIL (JJ2=1JJ2+1))>JNOH;                                               01376000    0018
WRITE(PRINT,FL11));                                                           01377000    0020
GO TO L0;                                                                     01378000    0023
L3: JTAUH=SVTAU1JJ2-1]+(SVTAU1JJ21-SVTAU1JJ2-1))*1JHS-SVHV1JJ2-1])/(SVHV      01379000    0024
```

```
(JJ2)=SVHV(JJ2-1));                                          01380000  0029
GO TO L8;                                                    01381000  0032
L2: JTAUH=SVTAU(JJ2);                                        01382000  0032
L8: JERRORS=JWHUA;                                           01383000  0034
SNSEARCH;                                                    01384000  0034
IF JERRORS<JWHUA THEN GO TO L340;                            01385000  0035
IF JNCR=JNSOREG THEN GO TO L7;                               01386000  0036
WRITE(PRINT,FL6);                                            01387000  0037
GO TO L0;                                                    01388000  0041
L7: JREFL=0;                                                 01389000  0041
L10: IF (XPR=(JNPART-JNSP))>0 THEN GO TO L70 ELSE IF XPR<0 THEN GO TO    01390000  0042
 L60;                                                        01391000  0046
SRAVGAGE;                                                    01392000  0046
IF JNH1ST<JNHMAX THEN GO TO L60;                             01393000  0047
GO TO L0;                                                    01394000  0048
L60: SRANGLE;                                                01395000  0048
IF JERRORS<JWHUA THEN GO TO L340;                            01396000  0049
JNSP=0;                                                      01397000  0050
L70: JNH1ST=JNH1ST+1;                                        01398000  0051
JNREFL=1;                                                    01399000  0053
JLOC=10;                                                     01400000  0054
JNSP=JNSP+1;                                                 01401000  0054
JH1=0;                                                       01402000  0056
JTAUH2=JTAUH;                                                01403000  0056
JH1=JHS;                                                     01404000  0057
JNCR=JNSOREG;                                                01405000  0058
JCOTH1=SVSANG(JNSP);                                         01406000  0059
JSITH1=SQRT(1-JCOTH1*JCOTH1);                                01407000  0060
JCPH11=1;                                                    01408000  0062
JSPH11=0;                                                    01409000  0063
JWAIT=SVWEIGHT(JNSP);                                        01410000  0064
JCAPH1=1;                                                    01411000  0065
```

```
JSAPH1=0;                                                    01412000  0065
JNCOL=1;                                                     01413000  0066
IF JIDUMP=0 THEN GO TO L80;                                  01414000  0067
WRITE(PRINT,FL78,LIST1);                                     01415000  0068
L80: JLOC=20;                                                01416000  0072
JR=JR1;                                                      01417000  0073
JH=JH1;                                                      01418000  0074
JREFL=0;                                                     01419000  0075
JTAUM1=JTAUM2;                                               01420000  0076
JCUTH=JCUTH1;                                                01421000  0078
JSITH=JSITH1;                                                01422000  0077
JCPH1=JCPHI1;                                                01423000  0078
JSPH1=JSPH11;                                                01424000  0079
JNCR1=JNCH;                                                  01425000  0079
JNCH=SVMATT(JNCH);                                           01426000  0080
IF JTDUMP=0 THEN GO TO LTOU;                                 01427000  0081
WRITE(PRINT,FL96,LIST2);                                     01428000  0082
L100: SHPATH;                                                01429000  0086
IF JERRORS<JWHUA THEN GO TO L340;                            01430000  0087
IF JPL>0 THEN GO TO L110;                                    01431000  0088
WRITE(PRINT,FL106,LIST3);                                    01432000  0090
JWHOA=JWHO4+1;                                               01433000  0093
GO TO L340;                                                  01434000  0095
L110: JT=JPL;                                                01435000  0097
JRHOT=0;                                                     01436000  0097
JOT=0;                                                       01437000  0098
JSUMOST=0;                                                   01438000  0099
JHT=JH;                                                      01439000  0100
L130: SHOSTH;                                                01440000  0100
IF JERRORS<JWHUA THEN GO TO L340;                            01441000  0101
IF JNCR=0 THEN GO TO L140;                                   01442000  0102
WRITE(PRINT,FL136);                                          01443000  0104
```

```
GO TO L0;                                                    01444000  0107
L140: JSUMOST=JSUMOST+JDIST;                                 01445000  0108
JLOC=50;                                                     01446000  0109
IF JTDUMP=0 THEN GO TO L144;                                 01447000  0110
WRITE(PRINT,FL142,LIST4);                                    01448000  0111
L144: IF JSUMOST>JT THEN GO TO L250;                         01449000  0115
JNCH=SVMAT[JNCH];                                            01450000  0116
JH=JH+JCOTH*JDIST;                                           01451000  0117
JTS=JDIST*JSITH;                                             01452000  0119
JRT=SQRT(JH*JH+JTS*JTS+2*JH*JTS*JCPHI);                      01453000  0120
IF JRT>JSMVAL THEN GO TO L550;                               01454000  0125
JCPHI=1;                                                     01455000  0126
JSPHI=0;                                                     01456000  0127
GO TO L600;                                                  01457000  0128
L550: JCPHI=(JTS+JR*JCPHI)/JRT;                              01458000  0130
JSPHI=JR*JSPHI/JRT;                                          01459000  0132
L600: JH=JRT;                                                01460000  0134
JNCH=JNCH;                                                   01461000  0134
JLOC=60;                                                     01462000  0135
IF JTDUMP=0 THEN GO TO L150;                                 01463000  0136
WRITE(PRINT,FL147,LISTS);                                    01464000  0137
L150: IF SVNROUND[JNCH]=0 THEN GO TO L170;                   01465000  0141
JH2=JH-2*JDELTA*JCOTH;                                       01466000  0143
 JJ2=2;                                                      01466100  0145
 DO BEGIN                                                    01466200  0146
 IF (XPH-(JH2-SVHV[JJ2]))<0 THEN GO TO L1800;                01466300  0148
 END UNTIL (JJ2=(JJ2+1))>JNUM;                               01466400  0149
 JJ2=JNUM;                                                   01466500  0151
 L1800: JTAUH2=SVTAU[JJ2-1]+(SVTAU[JJ2]-SVTAU[JJ2-1])*       01466600  0152
       (JH2-SVHV[JJ2-1])/(SVHV[JJ2]-SVHV[JJ2-1]);            01466700  0154
JH2=JR-2*JDELTA*JSITH*JCPHI;                                 01467000  0159
IF JNCR#1 THEN GO TO L1600;                                  01468000  0162
```

```
JNREFL=JNREFL+1;                                        01469000  0163
IF JNREFL=JNMAXR<1 THEN GO TO L1600;                    01470000  0164
JNMAXR=JNMAXR+1;                                        01471000  0166
GO TO L10;                                              01472000  0167
L1600: JREFL=1;                                         01473000  0168
JNRR=JNCR;                                              01474000  0168
JJA1L=SVJREFLT1JNRR];                                   01475000  0169
GO TO SWGU1[JJA1L];                                     01476000  0170
L161: JCOTH2=1;                                         01477000  0172
GO TO L166;                                             01478000  0174
L165: JCOTH2=1;                                         01479000  0174
L166: JS1TH2=0;                                         01480000  0175
JCPH12=JCPH1;                                           01481000  0176
JSPH12=JSPH1;                                           01482000  0177
JWA1T=JW41T=1$VALBF001JNCR1+SV513NOT[JNCR]*JCOTH1;      01483000  0178
GO TO L260;                                             01484000  0181
L170: JMPNEG=SVMPR1JJ1,JNCH];                           01485000  0181
SHSEARCH;                                               01486000  0183
IF JERRUNS<JWHUA THEN GO TO L340;                       01487000  0184
IF JNCR>0 THEN GO TO L180;                              01488000  0185
WRITE(PRINT,FL177,L1ST61;                               01489000  0186
GO TO L0;                                               01490000  0190
L180: JNCH2=JNCH;                                       01491000  0191
IF SVEMP[JNCR212SVEMP[JNCR1] THEN GO TO L188;           01492000  0191
SHRAND4(J1RASA,JRR);                                    01493000  0193
IF JRR>(SVEMP[JNCR21/SVEMP[JNCH111 THEN GO TO L310;     01494000  0194
JWA1T=JW41T=(SVEMP1JNCR1]/SVEMP[JNCR2]];                01495000  0196
GO TO L188;                                             01496000  0199
L310: SVNR1C01JNCR2]=SVNR1C01JNCR21+1;                  01497000  0199
JNRSTOP=JNRSTOP+1;                                      01498000  0202
GO TO L10;                                              01499000  0203
L188: JOT=JUT=JO1ST;                                    01500000  0203
```

```
GO TO L130;                                                    01501000  0205
L250: JD1ST=JT-JOT;                                            01502000  0205
JH2=JH+JCUTH*JD1ST;                                            01503000  0207
JTS=JD1ST*JS1TH;                                               0150A000  0209
JRT=SQRT(JR*JR+JTS*JTS+2*JH*JTS*JCPH1);                        01505000  0210
IF JRT>JSHVAL THEN GO TO L257;                                 01506000  0215
JCPH12=1;                                                      01507000  0216
JSPH12=0;                                                      01508000  0217
GO TO L258;                                                    01509000  0218
L257: JCPH12=(JTS+JR*JCPH1)/JRT;                               01510000  0220
JSPH12=JR*JSPH1/JRT;                                           01511000  0222
L258: JH2=JRT;                                                 01512000  0224
JCUTH2=JCOTH;                                                  01513000  0224
JS1TH2=JS1TH;                                                  01514000  0225
JFRACT=1/(JH2-SVHV(JJHR))/(SVHV(JJHT)-SVHV(JJHR));             01515000  0226
JSHAT10=SVSCATH(JJHB)+(SVSCATR(JJHT)-SVSCATR(JJHR))*JFRACT;    01516000  0229
JRATLEF=SVRAYR(JJHB)+(SVRAYR(JJHT)-SVRAYR(JJHB))*JFRACT;       01517000  0232
JHA1T=JHA1T*JSHAT10;                                           01518000  0235
L260: JNCH2=JNCH;                                              01519000  0236
JLUC=70;                                                       01520000  0237
JCUAZI=JCAPH1*JCPH12+JSAPH1*JSPH12;                            01521000  0238
JSUAZI=JSAPH1*JCPH12-JCAPH1*JSPH12;                            01522000  02A0
IF JIOUMPSO THEN GO TO L268;                                   01523000  02A3
WRITE(PRINT,FL26A,LIST7);                                      01524000  02AA
L268: SHDETECT;                                                01525000  02A8
IF JERRORS<JMHUA THEN GO TO L340;                              01526000  02A8
JNCOL=JNCOL+1;                                                 01527000  02A9
      JNOGO=JNOGO+1;                                           01527001  0251
IF JNCOL≤JNCMAX THEN GO TO L320;                               01528000  0252
JMAXCOL=JMAXCOL+1;                                             01529000  0253
GO TO L10;                                                     01530000  025A
L320:                                                          01531000  0255
```

```
SHSCTANG)                                                    01532000   0256
IF JENRURS<JWHOA THEN GO TO L340)                            01533000   0256
JH1=JH2)                                                     01534000   0257
JH1=JH2)                                                     01535000   0258
JNCR=JNCR2)                                                  01536000   0259
IF JWA11>JNCO THEN GO TO L80)                                01537000   0260
JNWAIT=JNWAIT+1)                                             01538000   0261
GO TO L10)                                                   01539000   0262
L340: IF JWHOA>JFLIM THEN GO TO L0)                          01540000   0263
JENROKS=JWHOA)                                               01541000   0264
GO TO L10)                                                   01542000   0265
L0:  END END)                                                01543000   0265
                                                          00A1 IS 0267 LONG, NEXT SEG 0039
                                                          0039 IS 0110 LONG, NEXT SEG 0007
PROCEDURE SKINPUT)                                           015AA000   03IA
BEGIN                                                        015A5000   03IA
OWN INTEGER DXI,DX2)                                         01558000   03IA
                                                          START OF SEGMENT ********** 0042
INTEGER JI1,JI2,JI3,JI4,JICHECK,JJATL,JLIS1,JLIS2,JI,JJ)                 0000
COMMENT  THE FOLLOWING PROCEDURES ARE USED:  SRCHECK)        01567000   0000
FORMAT FL10(5110),                                           01568000   0000
                                                          START OF SEGMENT ********** 0043
FL230(315,R5.2,R15),                                         01573000   0000
F(410(A110),                                                 01575000   0000
FLR10(A110),                                                 01577000   0000
FL2(X2,"PRODUCT OF NAZA AND NOMAX HAS EXCEEDED A0"/X2,       01578000   0000
 "JOB IS TERMINATED"),                                       01579000   0000
FL405(/                                                      015R0000   0000
 " THE NUMBER OF HISTORIES WAS NOT EQUALLY DIVISIBLE BY  THE NUMB",  015R1000   0000
 "ER OF DEVIATION GROUPS."/"  THE NUMBER OF HISTORIES WAS  RESET T",I6)  015R2000   0000
,                                                            015R3000   0000
FL510(2H10,7),                                               015R3100   0000
```

```
FL310(3H10.7,/10,H10.7),                                                0000
FL1701H10.7),                                                           0000
FL110(2110,H10.7),                                                      0000
>:130(6H10.7),                                                          0000
FL210,2110,H10.7),                                                      0000
FL920(/* INPUT NUMBER OF MATERIALS DOES NOT AGREE WITH NMAT. *),        01584000  0000
FL950(/* INPUT NUMBER OF BOUNDARIES DOES NOT AGREE WITH NBMAX.*),       01585000  0000
FL980(/* INPUT NUMBER OF REGIONS DOES NOT AGREE WITH NRMAX.*),          01586000  0000
FL1010(/* INPUT NUMBER OF DETECTORS DOES NOT AGREE WITH NDMAX.*),       01587000  0000
FL1040(/* INPUT NUMBER OF POINT COLLISIONS DOES NOT AGREE WITH NPCOL.*), 01588000  0000
FL1070(/* INPUT NUMBER OF POINT COSINES DOES NOT AGREE WITH NPA.*),     01589000  0000
FL2000(/                                                                01590000  0000
 * INPUT NUMBER OF REFLECTION BOUNDARIES DOES NOT AGREE WITH NREFL*,*B.*)  01591000  0000
,                                                                       01592000  0000
FL2030(/* INPUT SOURCE ANGLE OPTION DOES NOT AGREE WITH NAUP.*),        01593000  0000
FL2060(/* INPUT NUMBER OF SOURCE ANGLES DOES NOT AGREE WITH NAG.*),     01594000  0000
FL330(* HD(J) IS GREATER THAN HV(NDH)  FOR  J2= *,14,*.*)               01595000  0000
                                                                        0043 IS 0225 LONG, NEXT SEG 0042
LIST LIST1(JLIHRAY,JI1,JI2,JI3,JI4)$                                    01596000  0000
LIST LIST2(FOR DX1=1 STEP 1 UNTIL JNDH DO (SVHV[DX1],SVTAU[DX1],SVSCATH[  01597000  0010
DX1],SVRAYH[DX1]))$                                                     01598000  0016
LIST LIST3(SVNUFCOS[JI1],SVNPHANG[JI1],SVRAYLEE[JI1])$                  01599000  0024
LIST LIST4(FOR DX1=1 STEP 1 UNTIL JLIS1 DO SVD1FCOS[DX1,JII])$          01600000  0033
LIST LIST5(FOR DX1=1 STEP 1 UNTIL JLIS1 DO SVPDCOS[DX1,JII])$           01601000  0043
LIST LIST6(FOR DX1=1 STEP 1 UNTIL JLIS2 DO SVPHANG[DX1,JII]),           01602000  0053
LIST LIST7(FOR DX1=1 STEP 1 UNTIL JII DO (SVNRDUND[DX1],SVITYPE[DX1],SVC  01603000  0063
DFE[DX1]))$                                                             01604000  0069
LIST LIST8(FOR DX1=1 STEP 1 UNTIL JI2 DO (SVNREG[DX1],SVNRI[DX1],SVMAT[  01605000  0075
 DX1],SVEMP[DX1],FOR DX2=1 STEP 1 UNTIL 4 DO (SVIBF[DX2,DX1],SVMPR[DX2,  01606000  0081
 DX1]))))$                                                              01607000  0088
LIST LIST9(FOR DX1=1 STEP 1 UNTIL JII DO (SVHD[DX1],SVHD[DX1],SVAZD[DX1]  01608000  0097
,SVNPH[DX1],SVDBSS[DX1]))$                                              01609000  0103
```

```
LIST LIST10(FOR OX1:=1 STEP 1 UNTIL JI1 DO SVINCOLI[OX1]);                    01610000  0112
LIST LIST11(FOR OX1:=I STEP 1 UNTIL J12 DO SVCIPAI[OX1]);                     01611000  0121
LIST LIST12(FOR OX1:=I STEP 1 UNTIL JNAZA DO SVCAZA[OX1]);                    01612000  0130
LIST LIST13(SVALREQD[J11],SVSIGNDT[JI1]);                                     01613000  0139
LIST LIST14(FOR OX1:=1 STEP 1 UNTIL JI3 DO SVRFANGI[OX1,JII]);                01614000  0146
LIST LIST15(FOR OX1:=1 STEP 1 UNTIL J13 DO SVPOR[OX1,JI1]);                   01615000  0156
LIST LIST16(FOR OX1:=I STEP 1 UNTIL JI4 DO SVRFLCOSI[OX1,JIII]);              01616000  0166
LIST LIST17(FOR OX1:=I STEP 1 UNTIL JI2 DO SVCANG[OX1]);                      01617000  0176
LIST LIST18(FOR OX1:=1 STEP 1 UNTIL J12 DO SVPAG[OXI]);                       01618000  0185
LIST LIST19(FOR OX1:=I STEP 1 UNTIL JI2 DO SVMAGI[OXI]);                      01619000  019A
LIST LIST20(FOR OX1:=1 STEP 1 UNTIL JI3 DO SVSAZA[OX1]);                      01620000  0203
LIST LIST21(FOR OX1:=1 STEP 1 UNTIL J13 DO SVPAZA[OX1]);                      01621000  0212
LIST LIST22(JMS,JOLONG,JOFLIA,JSMVAL,JNCO,JELIM,JOMIN,FOR                               0221
 JINDEXI:=I STEP 1 UNTIL 1 DO SVAIJINDEXI);                                             0231
LIST LIST23(JNHMAX,JNGROUP,JNHMAX,JNRMAX,JNCMAX,JNDMAX,JNPA,JNPCOL,           01624000  0238
 JNAOP,JNAG,JNRFLB,JNMAT,JNSONEG,JMAXR,JIBASE,JIBAS1,JIBAS2,JIBAS3,            01625000  0249
 JIBASA,JIBAS5);                                                              01626000  0262
LIST LIST24(JNHMAX);                                                          01627000  0267
LIST LIST25(JJ);                                                              01628000  0272
BEGIN                                                                         01629000  0277
LABEL L5,L800,L700,L600,L500,L400,L300,L200,L100,L50,L900,L3000,L105,         01630000  0277
                                                          START OF SEGMENT ********** 004A
 L106,L107,L505,L506,L507,L520,L615,L111,L908,L930,L960,L990,L1020,           01631000  0000
 L1050,L1080,L2010,L2040,L2070,L2087,L350,L340,L360,L0;                       01632000  0000
SWITCH SWG01:=L800,L700,L600,L500,L400,L300,L200,L100,L50,L900,L3000;          01633000  0000
SWITCH SWG02:=L5,L520,L5,L520;                                                01634000  0006
JNMATP:=0;                                                                    01635000  0014
JNHMAXP:=0;                                                                   01636000  0015
JNHMAXP:=0;                                                                   01637000  0016
JNRFLHP:=0;                                                                   01638000  0017
JNDMAXP:=0;                                                                   01639000  0018
JNMCOLP:=0;                                                                   01640000  0016
```

```
JNMAP←0;                                                    016A1000  0019
JMAGE←0;                                                    01642000  0020
L5: JNOGO←0;                                                01643000  0021
READ(CARD,FL10,LIST1)IFINIS;                                016AA000  0021
GO TO SWGO1[JLIBRAY];                                       016A3000  0026
L50: JNOH←JI1;                                              01646000  0028
READ(CARD,FL170,LIST2)IFINIS;                               016A7000  0029
GO TO L5;                                                   016A8000  0034
L100: JNMATP←JNMATP+1;                                      016A9000  0038
SVMATERL[JNMATP]←JI1;                                       01650000  0039
JI←1;                                                       01651000  0040
DO BEGIN                                                    01652000  0041
  IF SVMATENL[JI1≠SVMATENL[JNMATP] THEN GO TO L105;         01653000  00A1
  IF JI≠JNMATP THEN GO TO L106;                             0165A000  00A3
  L105: END UNTIL (JI←(JI+1))>JNMATP;                       01655000  0044
GO TO L107;                                                 01656000  00A7
L106: JNMATP←JNMATP-1;                                      01657000  0047
L107: READ(CARD,FL110,LIST3)IFINIS;                         01658000  00A9
JLIS1←SVNDFCDS[JI1];                                        01659000  0055
JLIS2←SVNPHANG[JI1];                                        01660000  0056
IF SVRAYLEF[JI1]≥1 THEN GO TO L5;                           01661000  0057
READ(CARD,FL130,LISTA)[FINIS];                              01662000  0058
READ(CARD,FL130,LIST5)IFINIS;                               01663000  0063
READ(CARD,FL130,LIST6)IFINIS;                               0166A000  0068
GO TO L5;                                                   01665000  0073
L200: JNRHAXP←JI1;                                          01666000  0076
JNKHAXP←JI2;                                                01667000  0078
READ(CARD,FL210,LIST7)[FINIS];                              01668000  0079
READ(CARD,FL230,LIST8)[FINIS];                              01669000  008A
GO TO L5;                                                   01670000  0089
L300: JNOHAXP←JI1;                                          01671000  0092
READ(CARD,FL310,LIST9)[FINIS];                              01672000  0092
```

```
JJ+1);                                                          01673000  0097
DO BEGIN                                                        01674000  0098
  SVAZD1JJ1+SVAZD1JJI=.01745329 END UNTIL (JJ+(JJ+1))>JI1);     01675000  0098
GO TO L5);                                                      01676000  0102
L400: JNPCOLP+JI1);                                             01677000  0106
JNPAP+JI2);                                                     01678000  0106
READ(CARD,FL810,LIST10)1FINIS);                                 01679000  0107
READ(CARD,FL130,LIST1111FINIS);                                 01680000  0112
JNAZA+JI3);                                                     01681000  0117
READ(CARD,FL130,LIST12)1FINIS);                                 01682000  0118
GO TO L5);                                                      01683000  0123
L500: JNRFLBP+JNRFLHP+1);                                       01684000  0127
SVJREFLT1JI1)+J121                                              01685000  0128
SVNREFR(JNRFLBP)+JI1);                                          01686000  0129
J1+1);                                                          01687000  0130
DO BEGIN                                                        01688000  0131
  IF SVNREFR(J11#SVNREFR1JNRFLBP) THEN GO TO L505);             01689000  0131
  IF JI#JNRFLBP THEN GO TO L5061                                01690000  0133
  L505: END UNTIL (J1+(JI+1))>JNRFLBP1                          01691000  0134
GO TO L507);                                                    01692000  0137
L506: JNRFLBP+JNRFLHP+1);                                       01693000  0137
L507: READ(CARD,FL510,LIST1J11FINIS1);                          01694000  0139
JJA1L+SVJREFLT(JI11);                                           01695000  0145
GO TO SWG021JJA1L1);                                            01696000  0146
L520: SVNREANG1JI11+J1);                                        01697000  0148
READ(CARD,FL130,LIST1411FINIS);                                 01698000  0149
READ(CARD,FL130,LIST15)1FINIS);                                 01699000  0154
SVNREGOS1JI1)+JI4);                                             01700000  0159
READ(CARD,FL130,LIST1A)1FINIS);                                 01701000  0160
GO TO L5);                                                      01702000  0165
L600: JNAOPP+JI11                                               01703000  0170
JNAGP+JI21                                                      01704000  0170
```

```
JNSAZA+JI3);                                                    01705000  0171
READ(CARD,FL130,LIST17)[FINIS]);                                01706000  0172
READ(CARD,FL130,LIST18)[FINIS]);                                01707000  0177
IF JNAOPPSO THEN GO TO L615);                                   01708000  0182
READ(CARD,FL130,LIST19)[FINIS]);                                01709000  0183
L615: READ(CARD,FL130,LIST20)[FINIS]);                          01710000  0188
READ(CARD,FL130,LIST21)[FINIS]);                                01711000  0194
JJ+1);                                                          01712000  0199
DO BEGIN                                                        01713000  0199
  SVSAZA(JJ)+SVSAZA[JJ]*.01745329 END UNTIL (JJ+(JJ+1))>JI3);   01714000  0199
GO TO L5);                                                      01715000  0204
L700: READ(CARD,FL130,LIST22)[FINIS]);                          01716000  0211
GO TO L5);                                                      01717000  0216
L800: READ(CARD,FL810,LIST23)[FINIS]);                          01718000  0218
GO TO L5);                                                      01719000  0223
L900: JNPROB+J11);                                              01720000  0225
JI11+JNAZA*JNOMAX);                                             01721000  0225
IF JIIISAO THEN GO TO L111);                                    01722000  0227
WRITE(PRINT,FL2));                                              01723000  0226
WRITE(PRINT,FL2));                                              01724000  0231
ERROR(0));                                                      01725000  0235
L111: JIOUMP+JI2);                                              01726000  0236
JICHECK+JI3);                                                   01727000  0236
JNPART+JNHMAX DIV JNGROUP);                                     01728000  0237
IF JNHMAX=JNPART*JNGROUP THEN GO TO L908);                      01729000  0236
JNHMAX+JNPART*JNGROUP);                                         01730000  02A0
WRITE(PRINT,FL905,LIST2A));                                     01731000  02A1
L908: IF JNMATP=JNMAT THEN GO TO L930);                         01732000  02A5
WRITE(PRINT,FL920));                                            01733000  02A7
JNUGO+JNOGO+1);                                                 01734000  0250
L930: IF JNAMAXP=JNAMAX THEN GO TO L960);                       01735000  0252
WRITE(PRINT,FL950));                                            01736000  0253
```

```
JNUGO=JNOGO+1;                                                    01737000  0256
L980: IF JNRMAXP=JNRMAX THEN GO TO L990;                          01738000  0258
WRITE(PRINT,FL980);                                               01739000  0259
JNUGO=JNOGO+1;                                                    01740000  0262
L990: IF JNCMAXP=JNOMAX THEN GO TO L1020;                         01741000  0264
WRITE(PRINT,FL1010);                                              01742000  0265
  JUGO=JNOGO+1;                                                   01743000  0268
L1020: IF JNPCOLP=JNPCOL THEN GO TO L1050;                        01744000  0270
WRITE(PRINT,FL1040);                                              01745000  0271
JNUGO=JNUGO+1;                                                    01746000  0274
L1050: IF JNPAP=JNPA THEN GO TO L1080;                            01747000  0276
WRITE(PRINT,FL1070);                                              01748000  0277
JNUGO=JNOGO+1;                                                    01749000  0280
L1080: IF JNRFLMP=JNRFLA THEN GO TO L2010;                        01750000  0282
WRITE(PRINT,FL2000);                                              01751000  0283
JNUGO=JNOGO+1;                                                    01752000  0286
L2010: IF JNAUPP=JNAUP THEN GO TO L2040;                          01753000  0288
WRITE(PRINT,FL2030);                                              01754000  0289
JNUGO=JNOGO+1;                                                    01755000  0292
L2040: IF JNAGP=JNAG THEN GO TO L2070;                            01756000  0294
WRITE(PRINT,FL2060);                                              01757000  0295
JNUGO=JNOGO+1;                                                    01758000  0298
L2070: IF JNOGOP=0 THEN GO TO L5;                                 01759000  0300
IF JICHECK=0 THEN GO TO L2087;                                    01760000  0301
SNCHECK;                                                          01761000  0302
L2087: JJ1=2;                                                     01762000  0303
JJ=1;                                                             01763000  0303
DO BEGIN                                                          01764000  0304
  JJ2=JJ1;                                                        01765000  0304
  DO BEGIN                                                        01766000  0305
    IF (XPN=(SVMOD(JJ1=SVMVL,JJ2)))<0 THEN GO TO L350 ELSE IF XPR=0 THEN G  01767000  0305
O TO L340;                                                         01768000  0308
```

```
    END UNTIL (JJ2*(JJ2+1))>JNUM;                                    01769000    0304
   WRITE(PRINT,FL330,LIST25);                                        01770000    0311
   GO TO L3000;                                                      01771000    0315
   L350: SVTAUMD[JJ1*SVTAU[JJ2-11*(SVTAU[JJ21-SV1AU[JJ2-1])*(SVMD[   01772000    0315
    JJ]-SVMV[JJ2-1])/(SVMV[JJ2]-SVMV[JJ2-1]);                        01773000    0319
   GO TO L380;                                                       01774000    0324
   L380: SVTAUMD[JJ1*SVTAU[JJ2];                                     01775000    0324
   L380: JJ1*JJ2;                                                    01776000    0326
   END UNTIL (JJ2*(JJ+1))>JNUMAX;                                    01777000    0327
GO TO L01;                                                           01778000    0330
GO TO L5;                                                            01779000    0330
L3000: ERROR(0);                                                     01780000    0331
L01:  END END;                                                       01781000    0331
                                                                  0044 IS 0333 LONG, NEXT SEG 0042
                                                                  0042 IS 0299 LONG, NEXT SEG 0007
PROCEDURE MAINPROG;                                                  01782000    0314
BEGIN                                                                01783000    0314
COMMENT  THE FOLLOWING PROCEDURES ARE USED:  SRINPUT,SRMAIN,SRANSWER,01784000    0314
 SRDREAM;                                                            01785000    0314
LABEL L5;                                                            01786000    0314
                                                                  START OF SEGMENT ********** 0045
L5: SRINPUT;                                                         01787000    0000
SRMAIN;                                                              01788000    0000
SRANSWER;                                                            01789000    0001
SRDREAM;                                                             01790000    0001
GO TO L5;                                                            01791000    0002
 END;                                                                01792000    0002
                                                                  0045 IS 0003 LONG, NEXT SEG 0007
COMMENT  INITIALIZING BLOCKS;                                        01793000    0314
  XPH=0=K=0;                                                         01794000    0314
MAINPROG FINIS;                                                      01795000    0316
END; END;                                                            99999000    0317
```

```
                                                                    0007 IS 0320 LONG, NEXT SEG 0006
                                                                    0006 IS 0029 LONG, NEXT SEG 0002
          LKNJA+(TIME(2)=LKNJA)/60;OKV0K+(TIME(3)=OKV0K)/60;FZOVC+TIME(1);@LZAT;WR        99999100    C056
          ITE(PRINTIPAGE));WRITE(PRINI,CHGUB,100*LJLOU+GCPQV,LKNJA,OKV0K);END.          99999200    0064
                                                                    0002 IS C086 LONG, NEXT SEG C001

     ARCTAN IS SEGMENT NUMBER 00A6,PRT ADDRESS IS 0570
     COS IS SEGMENT NUMBER 00A7,PRT ADDRESS IS 0566
     EXP IS SEGMENT NUMBER 00A8,PRT ADDRESS IS 0110
     LN IS SEGMENT NUMBER 00A9,PRT ADDRESS IS 0116
     SIN IS SEGMENT NUMBER 0050,PRT ADDRESS IS 0567
     SQRT IS SEGMENT NUMBER 0051,PRT ADDRESS IS 0552
     OUTPUT(W) IS SEGMENT NUMBER 0052,PRT ADDRESS IS 004A
     OUTPUT(C) IS SEGMENT NUMBER 0053,PRT ADDRESS IS 0041
     INPUT(W) IS SEGMENT NUMBER 0054,PRT ADDRESS IS 0056
     INPUT(C) IS SEGMENT NUMBER 0055,PRT ADDRESS IS 0051
     GO TO SOLVER  IS SEGMENT NUMBER 0056,PRT ADDRESS IS 0053
     FILE CNTRL(W) IS SEGMENT NUMBER 0057,PRT ADDRESS IS 001A
     FILE CNTRL(C) IS SEGMENT NUMBER 0058,PRT ADDRESS IS 0015
     READ/WRITE IS SEGMENT NUMBER 0059,PRT ADDRESS IS 0018
NUMBER OF ERRORS DETECTED = 000.  COMPILATION TIME = 0226 SECONDS.
PRT SIZE=00A8;TOTAL SEGMENT SIZE=05937 WORDS;NUM STORAGE REQ.=06768 WORDS;NO. SEGS.=0059.
ESTIMATED CORE STORAGE REQUIREMENT = 11071 WORDS.
```

8.2 ALGOL Listing for LITE-II

The following is the ALGOL listing of LITE-II. Cards 1000 through 55000 were provided by the computing center at Fort Monmouth.

```
 BEGIN FILE OUT PRINT A (2,15)};INTEGER XRAZQ,VVUNU,FZOVC,LKNJA,OKVNK,QHA          1G00    0000
                                                                    START OF SEGMENT ********** 0002
N1,LJLOU,GCPOV};INTEGER AHRAY Z'KLA,QNCCL IO }I2]};OHHAT HHFHK ("TIME ON          2000    0005
                                                                    START OF SEGMENT ********** 0003
 ",IA,X9A,12,X1,A3," IQ",AZ),CHGUN ("TIME OFF ",I4,X30,"PHOC. TIME =",1I          3000    0007
0," SECS",X20,"1/0 TIME =",110," SECS")};DEFINE BLZAT =LJLOU +FZOVC OIV 2          4000    0007
                                                                    0003 IS 0028 LONG, NEXT SEG 0002
16000};GCPOV +FZOVC HOO 216000 /3A00 #};FILL ZIKLA [*]WITH 0,31,59,90,120,          5000    0007
                                                                    START OF SEGMENT ********** 0004
15),181,212,243,273,304 ,334,366}};ILL QNCCL [*]WITH 0,"JAN","FEH","MAH",          6000    000V
                                                                    0004 IS 0013 LONG, NEXT SEG 0002
                                                                    START OF SEGMENT ********** 0005
"APH","HAY","JUN","JUL","AUG","SEP","OCT","NOV","DEC"};ZOVC +TIME (I)};LK          7000    0010
                                                                    0005 IS 0013 LONG, NEXT SEG 0002
NJA +TIME (2)};OKVQK +TIME (3);VVUNU +TIME (0)};IF (IO=VVUNU,I1816)>VVUNU,          8000    0012
(2A1A])HOO 4 =0 THEN FOR XHAZQ =2 STEP 1 UNTIL 12 OU ZIKLA[XHAZQ]+ZIKLAI             9000    0017
XHAZQ]+1 JQHANI +100 HVVUNU ,(30 +0)+10 =VVUNU ,(34 +0)+VVUNU ,(A2 +A)};X          10000    0021
HAZQ +1};WHILE QHANI >ZIKLA [XRAZQ]OO XRAZQ +XRAZQ +1};QHANI +QHANI =ZIKLA          11000    0029
 [XHAZQ +1)};RLZAT};WHITE (PRINTIPAGE1,HHFRK,100=LJLOU+GCPOV,HNANI,QNCCL[X          12000    0033
HAZQ],VVUNU,[1A1,12)))                                                             13000    00AV
BEGIN                                                                              14000    0055
FILE IN CARD 0(2,10)1                                                              15000    0055
                                                                    START OF SEGMENT ********** 0006
FILE OUT PUNCH 0(2,10)}                                                            16000    0005
FIL1 XXXXXX 2(2,15)}                                                               17000    0010
FILE TAPE1 2(2,15)1                                                                18000    0015
FILE TAPE2 2(2,15)}                                                                19000    0020
FILE TAPE3 2(2,15)}                                                                20000    0025
FILE TAPE4 2(2,15)1                                                                21000    0030
FILE TAPE5 2(2,15)}                                                                22000    0035
FIL1 TAPE6 2(2,15)}                                                                23000    00A0
FILE TAPE7 2(2,15)1                                                                24000    00A5
```

```
FILE TAPE8 2(2,15);                                                    25000   0050
FILE TAPE9 2(2,15);                                                    26000   0055
FILE TAPE10 2(2,15);                                                   27000   0060
FILE TAPE11 2(2,15);                                                   28000   0065
FILE TAPE12 2(2,15);                                                   29000   0070
FILE TAPE13 2(2,15);                                                   30000   0075
FILE TAPE14 2(2,15);                                                   31000   0080
FILE TAPE15 2(2,15);                                                   32000   0085
FILE TAPE16 2(2,15);                                                   33000   0090
SWITCH FILE FILESW=XXXXXX,TAPE1,TAPE2,TAPE3,TAPE4,TAPE5,TAPE6,TAPE7,   34000   0095
   TAPE8,TAPE9,TAPE10,TAPE11,TAPE12,TAPE13,TAPE14,TAPE15,TAPE16;       35000   0107
LABEL  FINIS;                                                          36000   0110
REAL ARRAY  DATA[0:63,0:511];    COMMENT USED WITH DATA STATEMENTS ONLY;37000   0110
REAL Q,XPR;    INTEGER K;                                              38000   0120
FORMAT F(///////"STOP / PAUSE  NO. ",15), OCTL(2560);                 39000   0120
                                                  START OF SEGMENT ********** 0007
                                                  0007 IS 0017 LONG, NEXT SEG 0008
REAL PROCEDURE INT(ARG1);          VALUE ARG1;     REAL ARG1;          40000   0120
   INT=SIGN(ARG1)*ENTIER(ABS(ARG1));                                   41000   0120
REAL PROCEDURE TANH(ARG1);         VALUE ARG1;     REAL ARG1;          42000   0128
   TANH=((Q=EXP(ARG1*2))-1)/(Q+1);                                     43000   0128
REAL PROCEDURE MAX(ARG1,ARG2);     VALUE ARG1,ARG2; REAL ARG1,ARG2;    44000   0135
   MAX=IF ARG1>ARG2 THEN ARG1 ELSE ARG2;                               45000   0135
REAL PROCEDURE MIN(ARG1,ARG2);     VALUE ARG1,ARG2; REAL ARG1,ARG2;    46000   0140
   MIN=IF ARG1<ARG2 THEN ARG1 ELSE ARG2;                               47000   0140
REAL PROCEDURE DIM(ARG1,ARG2);     VALUE ARG1,ARG2; REAL ARG1,ARG2;    48000   0145
   DIM=MAX(ARG1-ARG2,0);                                               49000   0145
REAL PROCEDURE TSIGN(ARG1,ARG2);   VALUE ARG1,ARG2; REAL ARG1,ARG2;    50000   0149
   TSIGN=SIGN(ARG2)*ABS(ARG1);                                         51000   0149
REAL PROCEDURE LOG(ARG1);          VALUE ARG1;     REAL ARG1;          52000   015A
   LOG=LN(ARG1)/2.302585092498;                                        53000   015A
PROCEDURE ERROR(ARG1);             VALUE ARG1;     REAL ARG1;          54000   0160
```

```
BEGIN WRITE(PRINT,F,ARG1)) GO TO FINIS END                          55000  0160
REAL ARRAY                                                          56000  0170
     ARG[0:20],                                                     57000  0170
     SVTFLUX[0:10,     0:AU],                                       58000  0173
      SVFLUX[0:25,0:10,0:AU],                                       59000  0175
      SVDIFCOS[0:50,0:10 ],                                         60000  0178
      SVPOCOS [0:50,0:10 ],                                         61000  0180
      SVPHANG [0:50,0:10 ],                                         62000  0182
      SVAFLUX [0:25,0:10 ],                                         63000  018A
      SVPOR   [0:37,0:5  ],                                         64000  0187
      SVRFANG [0:37,0:5  ],                                         65000  0189
      SVSAFLUX[0:25,0: 10],                                         66000  0191
      SVSGFLUX[0:25,0:]0 ],                                         67000  0193
      SVFLUO  [0:]00,0:10 ],                                        68000  0196
      SVRFLCOS[0:50,0:10 ],                                         69000  0198
      SVA     [0:10 ],                                              70000  0200
      SVCANG  [0:37 ],                                              71000  0202
      SVFMP   [0:]00],                                              72000  0204
      SVFLUH  [0:10 ],                                              73000  0206
      SVC]PA  [0:30 ],                                              74000  0208
      SVFFLUX [0:]0 ],                                              75000  0210
      SVALMEOU[0:5  ],                                              76000  0212
      SVCOEE  [0:100],                                              77000  0214
      SVOVFLUX[0:10 ],                                              78000  0216
      SVHO    [0:10 ],                                              79000  0218
      SVPAG   [0:37 ],                                              80000  0220
      SVRAYLEE[0:10 ],                                              81000  0222
      SVSANG  [0:500],                                              82000  0224
      SVSTFLUX[0:]0 ],                                              83000  0226
      SVWEIGHT[0:500],                                              84000  0228
      SVORFLUX[0:10 ],                                              85000  0230
      SVPFANG [0:50 ],                                              86000  0232
```

```
      SVWAG   [0:37 ],                                    87000   0234
      SVPRFLT [0:50 1,                                    88000   0236
      SVRO    [0:10 ],                                    89000   0238
      SVRFLUX [0:10 ],                                    90000   0240
      SVRO[0:25,0:10],                                    91000   0242
      SVSIGNOT[0:10 ],                                    92000   0244
      SVSUMRHO[0:50 ],                                    93000   0246
      SVCRATIO [0:101 ,                                   94000   0248
      SVHV[0:1001,                                        95000   0250
      SVTAU[0:1001,                                       96000   0252
SVSCATR[0:1001,                                           97000   0254
          SVNRFH[0:100],                                  98000   0256
      SVRAYH[0:100],                                      99000   0258
      SVTAUHO[0:10],                                     100000   0260
          SVHOU[0:10,0:101 ,                             101000   0262
          SVCAZA[0:501 ,                                 102000   0265
          SVAZO[0:10] ,                                  103000   0267
          SVSAZA[0:37] ,                                 104000   0269
          SVPAZA[0:37] ,                                 105000   0271
          SVCCAZA[0:501 ,                                106000   0273
          SVANG[0:311 ,                                  107000   0275
      SVORSS  [0:10 1)                                   108000   0277
INTEGER ARRAY                                            109000   0278
      SV1B    [0:4 ,0:100],                              110000   0278
      SVMPH   [0:4 ,0:1001,                              111000   0281
      SVJREFLT[0:5  1,                                   112000   0283
      SVNDFCOS[0:10 ],                                   113000   0285
      SVNREG  [0:1001,                                   114000   0287
      SVINCOL [0:25 1,                                   115000   0289
      SVMAT   [0:1001,                                   116000   0291
      SVNR    [0:100],                                   117000   0293
      SVNPHANG[0:10 1,                                   118000   0295
```

```
      SVNRFANG(0:5  ),                                                  119000  0297
      SVNRICO (0:100),                                                  120000  0299
      SVITYPE (0:100),                                                  121000  0301
      SVMATERL(0:10 ),                                                  122000  0303
          SVIIRLE(0:50),                                                123000  0305
      SVNROUND(0:100),                                                  124000  0307
      SVNPHID (0:10 ),                                                  125000  0309
    NNFRIC(5),                                                          126000  0311
          SVNOETI(0:10) ,                                               127000  0313
          SVIIREF(0:50) ,                                               128000  0315
      SVNRFCOS(0:5  ));                                                 129000  0317
REAL                                                                    130000  0319
    JALPHA ,   JBETA ,   JBMAC ,   JCDEPH1,   JCDTH ,                   131000  0319
    JCDTH1 ,   JCDTH2 ,   JCMA ,   JCPH1 ,   JCPHI1 ,                   132000  0319
    JCPHI2 ,   JCPHIO ,   JCPRRO ,   JCPT ,   JCSA ,                    133000  0319
    JCSANG ,   JCTEP ,   JDELTA ,   JDEDH ,   JDIFH ,                   134000  0319
    JDIST ,   JDLONG ,   JDUH ,   JDT ,   JEAH ,                        135000  0319
    JELTH ,   JF1 ,   JFNPA ,   JFNRA ,   JH ,                          136000  0319
    JH1 ,   JH2 ,   JHS ,   JHT ,                                       137000  0319
                                JPAG ,   JPJH1 ,                        138000  0319
    JPL ,   JPSCAT ,   JR ,   JR1 ,   JR2 ,                             139000  0319
    JREFL ,   JRESULT,   JRHO ,   JRHOI ,   JRM ,                       140000  0319
    JRRO2 ,   JRRDSQ ,   JRT ,   JSDEPH1,   JSITH ,                     141000  0319
    JSITH1 ,   JSITH2 ,   JSMVAL ,   JSOD ,   JSPH1 ,                   142000  0319
    JSPH11 ,   JSPHI2 ,   JSPHID ,   JSPT ,   JSSANG ,                  143000  0319
    JSTEP ,   JSUMDST,   JSUMSQ ,   JT ,   JTEMP ,                      144000  0319
    JTS ,   JUPLMIT,   JWAIT ,   JWCD ,   JWHDA ,                       145000  0319
  JRATLEE, JTAUH, JTAUH1, JTAUH2,                                       146000  0319
          JCOAZI , JSOAZI , JCAPH1 , JSAPH1 ,                           147000  0319
          JAZMAX , JCOD , JSID , JSAM , JRAT , JANG ,                   148000  0319
          JCWAIT , JPAZ , JDIFANG , JCARK , JSPA , JCAP ,               149000  0319
          JARG , JAPA , JCAZAO , JADJUST , JPRI ,                       150000  0319
```

```
          JCAPHI1 , JSAPHI1 , JSRATIO ,                                    151000  0319
    JX    ,    JXR   ,    JERRORS,    JDMIN  )                             152000  0319
INTEGER                                                                    153000  0319
      JJMB, JJMT, JNREFL, JMAXR, JNMAXR, JIBAS1, JIBAS2,                   154000  0319
      JIBAS3, JIBAS4, JIBAS5, JNDM,                                        155000  0319
                                                  JIBASE ,                 156000  0319
    JICB  ,    JIDUMP ,    JJ1   ,    JKA1  ,    JKA2  ,                 157000  0319
    JKA3  ,    JKA4   ,    JLA   ,    JLR   ,    JLIBRAY,                158000  0319
    JLDC  ,    JLP    ,    JLSR  ,    JLST  ,    JMAI1 ,                 159000  0319
    JMAI2 ,    JMAXCOL,    JMPREG ,    JNAG  ,    JNAGP ,                 160000  0319
    JNAUP ,    JNAUPP ,    JNBMAX ,    JNRMAXP,    JNCB  ,                161000  0319
    JNCM  ,    JNCMAX ,    JNCOL  ,    JNCR  ,    JNCR1 ,                 162000  0319
    JNCR2 ,    JNCYC  ,    JNDFVG ,    JNDMAX ,    JNDMAXP,                163000  0319
    JNFORM ,   JNGROUP,    JNM1ST ,    JNMMAX ,    JNLB  ,                 164000  0319
    JNLM  ,    JNMAT  ,    JNMATP ,    JNDGO  ,    JNPA  ,                 165000  0319
    JNPAP ,    JNPART ,    JNPHASE,    JNPCOL ,    JNPCOLP,                166000  0319
    JNPROR ,   JNRA   ,    JNRFLB ,    JNRFLBP,    JNRING ,                167000  0319
          JNCAMD , JJJ , JJD ,                                             168000  0319
    JNRMAX ,   JNRMAXP,    JNMSTOP,    JNSOREG,    JNSY  ,                 169000  0319
       JLA7 , JJA00 , JJAOMAX , JKOUNT , JNRC2 ,                           170000  0319
          JNAZA0 , JJA0 , JIA0 , JMAXR1 , JJAMAX ,                         171000  0319
          JNAZA , JNSAZA , JINDEX1 , JI1I , JJ2 ,                          172000  0319
    JNSP  ,    JNDM   ,    JNWAIT ,        JNRB  )                         173000  0319
PROCEDURE SNRANDA(JIBASE,JRN);                                             174000  0319
INTEGER JIBASE;                                                            175000  0319
          REAL  JRN;                                                       176000  0319
BEGIN INTEGER A, B;                                                        177000  0319
                                                  START OF SEGMENT ********** 0008
      A.[12:16] ← JIBASE.[30:16];                                          178000  0000
      B.[12:35] ← JIBASE.[13:35];                                          179000  0002
      JIBASE.[12:36] ← A+B+JIBASE;                                         180000  0004
      A ← +0;                                                              181000  0007
```

```
      A.[21:27] * JIBASE.[11:27];                                  182000   0008
      JNN * A;                                                     183000   0010
      JNN * JNN/134217728.0;                                       184000   0011
END SNRANDA;                                                       185000   0013
                                                                 0008 IS 0017 LONG, NEXT SEG 0006
PROCFOUNE SRSEARCH;                                                186000   0319
BEGIN                                                              187000   0319
INTEGER   JI,JJ,JK;                                                188000   0319
                                                                 START OF SEGMENT ********** 0009
FORMAT FL23('* BOUNDARY',I3,' HAS BEEN INCORRECTLY IDENTIFIED.'),  189000   0000
                                                                 START OF SEGMENT ********** 0010
FL37('* POINT LIES ON BOUNDARY',I3),                               190000   0000
FL85('* SEARCH CYCLE THROUGH REGIONS IS NOT HANDELED PROPERLY.'),   191000   0000
FL95('* CANNOT FIND REGION FOR POINT WITH COORDINATES R = ',S1,E10.3,  192000   0000
 ', H = ',S1,E10.3);                                               193000   0000
                                                                 0010 IS 0054 LONG, NEXT SEG 0009
LIST LIST1(JNCH);                                                  194000   0000
LIST LIST2(JM,JN);                                                 195000   0005
LABEL L5,L10,L60,L50,L20,L25,L30,L35,L38,L40,L80,L90,L97,L0;        196000   0012
L5: JNSY=0;                                                        197000   0012
JNLR=JMPREG;                                                       198000   0013
JNUR=JNRMAX;                                                       199000   0014
L10: JK=JNLR;                                                      200000   0015
DO BEGIN                                                           201000   0016
  JJ=SVNR[JK];                                                     202000   0016
  JI=1;                                                            203000   0017
  DO BEGIN                                                         204000   0018
    JNCH=ABS(SV1R1J1,JK1);                                         205000   0018
    IF (XPR=(SV1TYPE1JNCH1=1))>0 THEN GO TO L30 ELSE IF XPR=0 THEN GO  206000   0020
     TO L25;                                                       207000   0023
    L20: WRITE(PRINT,FL23,LIST1);                                  208000   0024
    JNRUA=JNRUA+1;                                                 209000   0028
```

```
    GO TO L50;                                                    210000   0030
    L25: JXR=SVCOEF(JNCR)=JH;                                     211000   0030
    GO TO L35;                                                    212000   0032
    L30: JXR=SVCOEF(JNCR)=JH;                                     213000   0033
    L35: IF (XPH=(JXR))>0 THEN GO TO L40 ELSE IF XPR<0 THEN GO TO 214000   0034
     L38;                                                         215000   0037
    WRITE(PRINT,FL37,LIST1);                                      216000   0038
    JH=JH=JOFLT4=JCOTH;                                           217000   0041
    JH=JH=JOEL14=JSITH=JCPHI;                                     218000   0043
    GO TO L5;                                                     219000   0045
    L38: IF (XPR=(SVIB(JI,JK1))>0 THEN GO TO L40 ELSE IF XPH=0 THEN GO 220000   0046
     TO L20 ELSE GO TO L50;                                       221000   0050
    L40: IF (XPH=(SVIB(JI,JK)))<0 THEN GO TO L40 ELSE IF XPH=0 THEN GO 222000   0051
     TO L20;                                                      223000   0055
    L50: END UNTIL (JI=(JI+1))>JJ;                                224000   0056
  JNCR=JK;                                                        225000   0058
  GO TO L0;                                                       226000   0059
  L60: END UNTIL (JK=(JK+1))>JNUH;                                227000   0059
IF (XPR=(JNSY))>0 THEN GO TO L90 ELSE IF XPR<0 THEN GO TO L80;    228000   0062
JNSY=1;                                                           229000   0065
JNLH=1;                                                           230000   0066
JNUB=JMPREG;                                                      231000   0068
GO TO L10;                                                        232000   0067
L80: WRITE(PRINT,FL95);                                           233000   0068
JHHD8=JHHD8+1;                                                    234000   0071
GO TO L97;                                                        235000   0072
L90: WRITE(PRINT,FL95,LIST2);                                     236000   0073
JHHD8=JHHD8+1;                                                    237000   0077
L97: JNCR=0;                                                      238000   0079
L0:  END;                                                         239000   0079
                                                    0009 IS 0083 LONG, NEXT SEG 000A
PROCEDURE SRDSIN0;                                                240000   0319
```

```
BEGIN                                                          241000   0319
INTEGER      JJ,JK;                                            242000   0319
                                                  START OF SEGMENT ********** 0011
COMMENT  THE FOLLOWING PROCEDURES ARE USED:  SRSEARCH;         243000   0000
FORMAT FL15(/" BOUNDARY",F3," HAS BEEN IDENTIFIED INCORRECTLY."),  244000   0000
                                                  START OF SEGMENT ********** 0012
FL55(/" LOC =",I4,"  ICB =",I4,"  X =",F10.3,"  RMAC =",E10.3,     245000   0000
   "  D. T =",E10.3/" M =",E10.3,"  R =",E10.3,"  COEF((ICB) =",E10.3,  246000   0000
   "  ITYPE((ICM) =",I4),                                      247000   0000
FL75(/" COLLISION POINT IS WITHIN A DISTANCE OF 1.1 DELTA FROM BOUNDAR=",  248000   0000
"Y",I4,",  IT WAS MOVED OFF THE BOUNDARY,"!!                   249000   0000
                                                  0012 IS 0066 LONG, NEXT SEG 0011
LIST LIST1(JICM));                                             250000   0000
LIST LIST2(JLOC,JICB,JX,JRMAC,JD(ST,JM,JR,SVCOEF((JICB),SVITYPE((JICM));  251000   0005
LIST LIST3(JNCM);                                              252000   0021
LABEL L5,L60,L20,L30,L39,L36,L3R,L5R,L0;                       253000   0026
COMMENT  SUBROUTINE OSIMO;                                     254000   0026
JNCR=0;                                                        255000   0026
JJ=1;                                                          256000   0027
JLOC=105;                                                      257000   0028
L5: JOIST=JOLUNG;                                              258000   0028
JK=SVNR[JNCR];                                                 259000   0029
JJ=1;                                                          260000   0030
DO BEGIN                                                       261000   0031
  JICR=ABS(SVIR[JJ,JNCR]);                                     262000   0031
  IF (XPR=(SVITYPE[JICB]=1))>0 THEN GO TO L30 ELSE IF XPR=0 THEN GO TO  263000   0033
   L20;                                                        264000   0036
  WRITE(PRINT,FL15,LIST1);                                     265000   0037
  JWHO4=JWHO4+1;                                               266000   0041
  GO TO L0;                                                    267000   0042
  L20: IF ABS(JCOTH)<JSMVAL THEN GO TO L60;                    268000   0042
  JX=(SVCUEE[JICB]-JM)/JCOTH;                                  269000   0044
```

```
  GO TO L39;                                                    270000   0046
  L30: IF ABS(JSITH)≤JSHVAL THEN GO TO L60;                     271000   0047
  JRRAC=(SVCOEE(JICR)**2=(JRR*JSPHI)**2);                       272000   0048
  IF JRRAC≤0 THEN GO TO L60;                                    273000   0051
  IF (XPR=(SVCOEE(JICR)=JR))>0 THEN GO TO L38 ELSE IF XPR<0 THEN GO TO   274000   0052
   L36;                                                         275000   0056
  JMPREG=JNCR;                                                  276000   0056
  SRSEARCH;                                                     277000   0057
  IF JERRORS<JMMDA THEN GO TO L0 ELSE GO TO L5;                 278000   0057
  L36: JX=(=JR*JCPHI=SQRT(JRRAC))/JSITH;                        279000   0059
  GO TO L39;                                                    280000   0063
  L38: JX=(=JR*JCPHI+SQRT(JRRAC))/JSITH;                        281000   0065
  L39: IF JIDUMPS0 THEN GO TO L56;                              282000   0068
  WRITE(PRINT,FL55,LIST2);                                      283000   0070
  L56: IF JX≤0 THEN GO TO L60;                                  284000   0074
  IF JDIST≤JX THEN GO TO L60;                                   285000   0075
  JDIST=JX+JDELTA;                                              286000   0076
  JNCR=JICR;                                                    287000   0077
  JJ1=JJ;                                                       288000   0078
  L60: END UNTIL (JJ=(JJ+1))>JK;                                289000   0079
IF JDIST≥1.1*JDELTA THEN GO TO L0;                              290000   0082
WRITE(PRINT,FL75,LIST3);                                        291000   008A
JM=JM+JDELTA*JCOTH;                                             292000   0087
JR=JR+JDELTA*JSITH*JCPHI;                                       293000   0089
JMPREG=SVMPRIJJ1,JNCR);                                         294000   0091
SRSEARCH;                                                       295000   0093
IF JNCR>0 THEN GO TO L5;                                        296000   009A
L0:  END;                                                       297000   0095
                                                  0011 IS OIOI LONG, NEXT SEG 0008
PROCEDURE SHDETECT;                                             298000   0319
REMIN                                                           299000   0319
REAL    JCOD,  JSIO;  INTEGER  JJ,JK,JL,JM;                     300000   0319
```

```
                                                                    START OF SEGMENT ********** 0013
      INTEGER     JLC, J1, JJ3, JJ2;                                301000    0000
FORMAT FL190(" LOC =",I5," LA =",I4," LC=",I4," LP=",I4," NCR1=",I4/  302000    0000
                                                                    START OF SEGMENT ********** 0014
 " COTH2 =",S1,E10.3," 1 =",I4," M2 =",E13.4," MO(1) =",S1,E13.4),    303000    0000
FL2A0(" LOC= ",I4," J2 =",I4/" RESULT=",S1,E10.3," FLUX =",S1,E10.3,  304000    0000
 " FLUO =",S1,E13.3," AFLUX =",S1,E13.3," REFL =",S1,E10.3," AFLUX =", 305000    0000
 S1,E13.3)};                                                        306000    0000
                                                                    0014 IS 005A LONG, NEXT SEG 0013
LIST LIST1(JLOC,JLA,JLC,JLP,JNCR1,JCOTH2,JI,JM2,SVMO(J1))};         307000    0000
LIST LIST2(JLOC,JJ2,JRESULT,SVFLUX(JLA,JLP,JJA000),SVFLUOI,JNMC2,JJ2),SVMF 308000    0016
LUX(JJ2),J REL,SVAFLUXI,JLC,JJ2))};                                 309000    0026
LABEL L20,L12,L40,L60,L70,L100,L90,L120,L130,L160,L150,L170,L200,    310000    0034
 L250,L220,L0};                                                     311000    0034
JL=1};                                                              312000    0034
DO BEGIN                                                            313000    0035
  IF JCOTH2>SVCIPA[JL] THEN GO TO L20};                             314000    0035
  END UNTIL (JL=(JL+1))>JNMA};                                      315000    0036
L20: JLA=JL};                                                       316000    0039
COMMENT  DETERMINE INDEX , LAZ , FOR AZIMOTHAL TABLE};              317000    0039
JL=1};                                                              318000    0039
DO BEGIN                                                            319000    0040
  IF JCAPHI2SV[ /A[JL] THEN GO TO L12};                             320000    0040
  END UNTIL (JL=(JL+1))>JNAZA};                                     321000    0042
L12: JLAZ=JL};                                                      322000    0044
COMMENT  DETERMINE NUMBER OF COLLOSION PRINTOUT GROUP INDEX, LC,};  323000    0045
JM=1};                                                              324000    0045
DO BEGIN                                                            325000    0046
  IF SVINCOL[JM]>JNCOL THEN GO TO L40};                             326000    0046
  END UNTIL (JM=(JM+1))>JNPCOL};                                    327000    0048
L40: JLC=JM};                                                       328000    0050
COMMENT  DETERMINE NUMBER OF REFLECTION PRINTOUT GROUP INDEX, LP,}; 329000    0051
```

```
JM+1;                                                                  330000  0051
OO BEGIN                                                               331000  0052
  IF JM2JNREFL THEN GO TO L60;                                         332000  0052
  END UNTIL (JM+(JM+1))>JMAXR;                                         333000  0053
L60: JLP+JM1                                                           334000  0056
L70: JI+1;                                                             335000  0056
OO BEGIN                                                               336000  0057
  IF (XPR*(JM2-SVHO[JI1))<0 THEN GO TO L100 ELSE IF XPR=0 THEN GO TO   337000  0057
   L90;                                                                338000  0061
  END UNTIL (JI+(JI+1))>JNOMAX;                                        339000  0061
IF JCOTH2+JSMVAL<0 THEN GO TO L120 ELSE GO TO L0;                      340000  0063
L90: JH2+JM2+JDELTA*JCOTH2;                                            341000  0066
GO TO L70;                                                             342000  0067
COMMENT  H2 IS BELOW DETECTOR PLANE HO(I);                             343000  0068
L100: IF ABS(JCOTH2)<JSMVAL THEN GO TO L0;                             344000  0068
IF (XPR*(JCOTH2))>0 THEN GO TO L160 ELSE IF XPR=0 THEN GO TO L0 ELSE GO 345000  0070
 TO L130;                                                              346000  0073
COMMENT  FLUX IS CALCULATED FOR DETECTORS BELOW H2;                    347000  0074
L120: JJ3+JNOMAX;                                                      348000  007A
GO TO L150;                                                            349000  007A
L130: IF J1S1 THEN GO TO L0;                                           350000  0075
JJ3+JI-1;                                                              351000  0077
L150: JJ1+JI;                                                          352000  0078
GO TO L170;                                                            353000  0079
COMMENT  FLUX IS CALCULATED FOR DETECTOR PLANES ABOVE H2;              354000  0080
L160: JJ3+JNOMAX;                                                      355000  0080
JJ1+JI;                                                                356000  0081
L170: IF JTOUMP50 THEN GO TO L200;                                     357000  0082
JLOC+90;                                                               358000  0084
WRITE(PRINT,FL190,LIST1);                                              359000  0085
L200: JJ2+JJ1;                                                         360000  0088
OO BEGIN                                                               361000  0089
```

```
  JRESULT=JWAITREXP((JTAUM2-SVTAUMD[JJ2I])/JCOTM2)/ABS(JCOTM2))                    362000    0089
  JJADD=JLAZ+JNAZA*(JJ2-1))                                                        363000    0093
  SVFLUX[JLA,JLP,JJA001=SVFLUX[JLA,JLP,JJADDI+JRESULT)                             364000    0096
  SVFLUD[JNCM2,JJ2I=SVFLUD[JNCM2,JJ2I+JRESULT)                                     365000    0101
  SVAFLUX[JLC,JJ2)=SVAFLUX[JLC,JJ2I+JRESULT)                                       366000    010A
  IF JRELESO THEN GO TO L260)                                                      367000    0108
  SVRFLUX[JJ2I=SVRFLUX[JJ2)+JRESULT)                                               368000    0109
  SVROD[JLP,JJ2)=SVROD[JLP,JJ2)+JRESULT)                                           369000    0111
  L2201 JLDC=110)                                                                  370000    0114
  IF JIDUMPSO THEN GO TO L250)                                                     371000    0115
  WRITE(PRINT,FL240,LIST2))                                                        372000    0117
  L250: END UNTIL (JJ2=(JJ2+I))>JJ3)                                               373000    0120
L01  END)                                                                          374000    0123
                                                0013 IS 0130 LONG, NEXT SEG 0006
PROCEDURE SHANSWER)                                                                375000    0319
BEGIN                                                                              376000    0319
       INTEGER  JJN, JJJ)                                                          377000    0319
                                                START OF SEGMENT ********** 0015
REAL  JEGROUP,JENHMAX)     INTEGER  J1,JJ,JK,JN,JM )                               378000    0000
OWN INTEGER DX1)                                                                   379000    0000
FORMAT FL110(" RADIATION RESEARCH ASSOCIATES #LITE# PROBLEM",I10),                 380000    0000
                                                START OF SEGMENT ********** 0016
FL120(/" HISTORY TERMINATION COUNTERS."),                                          381000    0000
FL130(/" ",I9,                                                                     382000    0000
 " HISTORIES WERE TERMINATED WHEN THE COLLISION NUMBER EXCEEDED",I6,".",/          383000    0000
110," HISTORIES WERE TERMINATED BY THE REGION IMPORTANCE PARAMETERS."/             384000    0000
 I10," HISTORIES WERE TERMINATED BY MINIMUM WEIGHT CUTOFF."/110,                   385000    0000
 " HISTORIES WERE TERMINATED AFTER  MAXIMUM NUMBER OF REFLECTIONS",".",")),        386000    0000
FL135(/" ",I9," COLLISIONS OCCURRED."),                                            387000    0000
FL150(/                                                                            388000    0000
 " PARTICLES TERMINATED IN EACH REGION BY REGION IMPORTANCE PARAM",                389000    0000
 "ETERS."),                                                                        390000    0000
```

```
FL160(/                                                                 391000    0000
 * REGION HISTORIES   REGION HISTORIES   REGION HISTORIES   REGIO",      392000    0000
 "N HISTORIES"/                                                         393000    0000
 *        TERMINATED          TERMINATED          TERMINATED       ",    394000    0000
 *  TERMINATED"),                                                       395000    0000
FL170(" ",14,19,I10,I9,I10,I9,I10,I9),                                  396000    0000
FL190(/                                                                 397000    0000
 *  SCATTERED LIGHT INTENSITY VERSUS ANGLE AND NUMBER OF  REFLECT",     398000    0000
 "IONS FROM SURFACE ONE."),                                             399000    0000
FL210("  ANGLE",X33,"COLLISION"),                                       400000    0000
FL250(" (COSINE)",18,6(X9,I2)),                                         401000    0000
FL262(" (COSINE)    TOTAL"),                                            402000    0000
FL264(" ",X23,"TOTAL"),                                                 403000    0000
FL266(" ",X34,"TOTAL"),                                                 404000    0000
FL268(" ",X45,"TOTAL"),                                                 405000    0000
FL270(" ",X56,"TOTAL"),                                                 406000    0000
FL272(" ",X67,"TOTAL"),                                                 407000    0000
FL274(" ",X78,"TOTAL"),                                                 408000    0000
FL450(/                                                                 409000    0000
 *                    SCATTERED LIGHT INTENSITY VERSUS REGION OF ",     410000    0000
 "SCATTER"),                                                            411000    0000
FL460(/"  REGION   ",X30,"DETECTOR"),                                   412000    0000
FL485(/"               01"),                                            413000    0000
  .                                                                  0016 IS 025? LONG, NEXT SEG 0015
                                                                     START OF SEGMENT ********** 0017
FL495(/"               01         02"),                                 414000    0000
FL505(/"               01         02         03"),                      415000    0000
FL515(/"               01         02         03         04"),           416000    0000
FL525(/"               01         02         03         04         05"), 417000    0000
FL535(/                                                                 418000    0000
 *             01         02         03         04         05  ",       419000    0000
 *       06"),                                                          420000    0000
```

```
FL585(/                                                                 421000   0000
 "              01         02         03         04         05  ",      422000   0000
 "      06         07"),                                                423000   0000
FL605(/"              08"),                                             424000   0000
FL615(/"              08         09"),                                  425000   0000
FL625(/"              08         0          10"),                       426000   0000
FL680(" LIGHT SCATTERED FROM REFLECTION SURFACES TO EACH DETECTOR,"),   427000   0000
FL690(/" NO OF REFLECTIONS                        DETECTOR "),          428000   0000
FL705(/" ",X12,5(X9,I2)),                                               429000   0000
FL735(7HA,8,X10,I4,I2,I2,I2," ACC"),                                    430000   0000
FL191(/X10,"CUSINES OF AZIMUTHAL RANGE =  ",S1,E10.3," TO  ",S1,E10.3), 431000   0000
FL200(/"       SOURCE HEIGHT H=",S1,F10.3,                              432000   0000
 ",   DETECTOR COORDINATES  HO=",S1,E10.3," RO=,",S1,F10.3),            433000   0000
FL280(" ",R7,8,X1,S1,7E11.3),                                           434000   0000
FL300(/"  TOTAL  ",S1,7E11.3),                                          435000   0000
FL560("    ",12,X3,S1,7E11.3),                                          436000   0000
FL580(/"  TOTAL  ",S1,7E11.3),                                          437000   0000
FL710("       ",12,X9,S1,5E11.3),                                       438000   0000
FL720(/"     TOTAL        ",S1,5E11.3),                                 439000   0000
FL745(S1,6E11.4,A,I8,I2,I2,I2," ACC"),                                  440000   0000
FL747(S1,3E11.4,A,X33,I4,I2,I2,I2," ACC")$                              441000   0000
                                                                        0017 IS 0263 LONG, NEXT SEG 0015
LIST LIST1(JNPNUR)$                                                     442000   0000
LIST LIST2(JNMAXCOL,JNCMAX,JNRSTOP,JNMAIT,JNMAXH)$                      443000   0005
LIST LIST3(JNOGU)$                                                      444000   0015
LIST LIST4(FOR DX1=1 STEP 1 UNTIL JNRMAX DO (DX1,SVNRI(DX1)))$          445000   0020
LIST LIST5(JCAZAO,SVCC4ZA(JJA01))$                                      446000   0031
LIST LIST6(JNS,SVH01JJ1,SVH01JJ1)$                                      447000   0038
LIST LIST7(FOR DX1=JKA1 STEP 1 UNTIL JKA2 DO SV11KFE(DX1))$             448000   0046
LIST LIST8(SVC1PA(JN),FOR DX1=JKA1 STEP 1 UNTIL JKA2 DO SVFLUX1(JN,DX1,  449000   0055
 JJA00))$                                                               450000   0061
LIST LIST9(FOR DX1=JK41 STEP 1 UNTIL JK42 DO SVTFLUX10(X1,JJA00))$      451000   0067
```

```
LIST LIST10(SVNREG[JI],FOR DXI←1 STEP 1 UNTIL JNFORM DO SVFLUD[JI,DX1]))          452000   0077
LIST LIST11(FOR DXI←1 STEP 1 UNTIL JNFORM DO SVFLUR[DXI]))                        453000   0088
LIST LIST12(SVNREG[JI],FOR DXI←8 STEP 1 UNTIL JNFORM DO SVFLUD[JI,DX1]))          454000   0097
LIST LIST13(FOR DXI←8 STEP 1 UNTIL JNFORM DO SVFLUR[DXII))                        455000   0108
LIST LIST14(FOR DX1←JK81 STEP 1 UNTIL JK82 DO SVNOETIOXII))                       456000   0117
LIST LIST15(SVIIREE[JJJ],FOR DX1←JK81 STEP 1 UNTIL JK82 DO SVNDOI                 457000   0126
 JJJ,DX1)))                                                                       458000   0130
LIST LIST16(FOR DX1←JK81 STEP 1 UNTIL JK82 DO SVRFLUX[DX1]))                      459000   0137
LIST LIST17(FOR DX1←JK81 STEP 1 UNTIL JK82 DO SVANG[DX1],JNPRDR,JIAO,             460000   0146
 JJD,JNC4RD))                                                                     461000   0155
LIST LIST18(FOR DX1←JK81 STEP 1 UNTIL JK82 DO SVFLUX[JJN,DXI,JJJ),                462000   0160
 JNPRDR,JIAO,JJD,JNC4RD))                                                         463000   0168
BIGIN                                                                             464000   0176
LAHEL L998,L180,L185,L240,L261,L275,L263,L265,L267,L269,L271,L273,                465000   0176
                                                    START OF SEGMENT ********** 0018
 L14,L430,L440,L480,L490,L500,L510,L520,L530,L540,L600,L610,L620,                 466000   0000
 L550,L663,L650,L670,L700,L770,L730,L0)                                           467000   0000
SWITCH SWG01←L261,L263,L265,L267,L269,L271,L273,L275)                             468000   0000
SWITCH SWG02←L480,L490,L500,L510,L520,L530,L540,L600,L610,L620)                   469000   0007
COMMENT  SUHROUTINE ANSWER)                                                       470000   0016
JFNHM&X←JNHM&X)                                                                   471000   0016
JFGRROUP←JNGROUP)                                                                 472000   0017
JJAOM&X←JNDM&X←JN&Z&)                                                             473000   0018
JLST←JM&XR+1)                                                                     474000   0019
JJ←1)                                                                             475000   0021
DO BEGIN                                                                          476000   0021
  J1←1)                                                                           477000   0021
  DO BEGIN                                                                        478000   0022
    JK←I)                                                                         479000   0022
    DO BEGIN                                                                      480000   0023
      SVFLUX[JK,JI,JJ]←SVFLUX[JK,JI,JJI/JFNHM&X)                                  481000   0023
      SVFLUX[JK,JLST,JJ1←SVFLUX[JK,JLST,JJ1+SVFLUX[JK,JI,JJI)                     482000   0028
```

```
      SVTFLUX[JI,JJ]+SVTFLUX[JI,JJ]+SVFLUX[JK,JI,JJ];                    483000  0035
      END UNTIL (JK+(JK+11)>JNPA);                                       484000  0040
    SVTFLUX[JLST,JJ]+SVTFLUX[JLST,JJ]+SVTFLUX[JI,JJ];                    485000  0042
    SVTIREE(JI]+JI+1;                                                    486000  0047
    END UNTIL (JI+(JI+111>JMAXM END UNTIL (JJ+(JJ+111>JJAOMAX;           487000  0048
JJ+1;                                                                    488000  0053
DO BEGIN                                                                 489000  0054
  JI+1;                                                                  490000  0054
  DO BEGIN                                                               491000  0054
    SVRUO(JI,JJ1+SVNOOIJI,JJ1/JFNHM0X END UNTIL (JI+(JI+111>JNAXR END    492000  0054
 UNTIL 'JJ+(JJ+111>JNOM0X;                                               493000  0060
JJ+1;                                                                    494000  0062
DO BEGIN                                                                 495000  0063
  JM+1;                                                                  496000  0063
  DO BEGIN                                                               497000  0064
    SVFLUO(JM,JJI+SVFLUOIJM,JJ]/JFNHM0X;                                 498000  0064
    SVFLURIJJI+SVFLUN(JJI+SVFLUOIJM,JJI;                                 499000  0067
    END UNTIL (JM+(JM+111>JNHM0X;                                        500000  0070
  SVRFLUXIJJI+SVRFLUXIJJ1/JFNHM0X;                                       501000  0073
  END UNTIL (JJ+(JJ+111>JNOM0X;                                          502000  0075
COMMENT  SUBROUTINE RESULT;                                              503000  0077
WRITE(PRINT(PAGE));                                                      504000  0077
WRITE(PRINT,FL110,LIST1];                                                505000  0080
WRITE(PRINT,FL120];                                                      506000  0084
WRITE(PRINT,FL130,LIST2];                                                507000  0087
WRITE(PRINT,FL135,LIST3];                                                508000  0091
IF JNRSTOP=0 THEN GO TO L99A;                                            509000  0095
WRITE(PRINT,FL150];                                                      510000  0096
WRITE(PRINT,FL160];                                                      511000  0100
WRITE(PRINT,FL170,LIST4];                                                512000  0103
L99A; JROUNT+0;                                                          513000  0107
JJ+1;                                                                    514000  0108
```

```
DO BEGIN                                                    515000  0109
  SVCCAZA[JJ]+SVCAZA[JJ] END UNTIL (JJ+(JJ+1))>JNAZA);      516000  0109
JNAZAD+JNAZA);                                              517000  0113
L180: JJ+1;                                                 518000  0114
DO BEGIN                                                    519000  011A
  JJAD+(JJ-1)*JNAZAD;                                       520000  0114
  JCAZAU+1;                                                 521000  0116
  J1AD+1;                                                   522000  0117
  DO BEGIN                                                  523000  0118
    JJADD+JJAD+J1AD;                                        524000  0118
    JKA2+0;                                                 525000  0119
    JKA3+0;                                                 526000  0120
    L185: WRITE(PRINT[PAGE]);                               527000  0120
          WRITE(PRINT,FL110,LIST11);                        528000  012A
    WRITE(PRINT,FL190);                                     529000  0128
    WRITE(PRINT,FL191,LIST5);                               530000  0131
    WRITE(PRINT,FL200,LIST6);                               531000  0135
    WRITE(PRINT,FL210);                                     532000  0139
    JKA1+JKA2+1;                                            533000  01A2
    JKA2+JKA1+6;                                            534000  0143
    IF JKA2>JMAXR THEN GO TO L2A0;                          535000  01A5
    JKA3+1;                                                 536000  01A6
    JKA2+JMAXR;                                             537000  01A7
    IF JKA1>JMAXR THEN GO TO L261;                          538000  01A7
    L2A0: WRITE(PRINT,FL250,LIST7);                         539000  0149
    IF JKA3=0 THEN GO TO L275;                              540000  0152
    JKA2+JKA2+1;                                            541000  0154
    JKAA+JKA2-JKA1+1;                                       542000  0155
    GO TO SWG01[JKAA];                                      543000  0157
    L261: WRITE(PRINT,FL264);                               544000  0159
    GO TO L275;                                             545000  0162
    L263: WRITE(PRINT    ,FL264);                           546000  0163
```

```
    GO TO L275;                                                                  547000  0166
    L265: WRITE(PRINT    ,*)L266);                                               548000  0167
    GO TO L275;                                                                  549000  0170
    L267: WRITE(PRINT    ,*)L268);                                               550000  0171
    GO TO L275;                                                                  551000  0174
    L269: WRITE(PRINT    ,*)L270);                                               552000  0175
    GO TO L275;                                                                  553000  0178
    L271: WRITE(PRINT    ,*)L272);                                               554000  0179
    GO TO L275;                                                                  555000  0182
    L273: WRITE(PRINT    ,*)L274);                                               556000  0183
    L275: JN=1;                                                                  557000  0186
    DO BEGIN                                                                     558000  0187
      WRITE(PRINT,FL280,LIST8);                                                  559000  0187
      ;    UNTIL (JN=(JN+1))>JNMA;                                               560000  0191
    WRITE(PRINT,FL300,LIST9);                                                    561000  0193
    IF JK=350 THEN GO TO L185;                                                   562000  0197
    JCA2A0=SVCCA2A[J1A0];                                                        563000  0198
    END UNTIL (J1A0=(J1A0+1))>JN=2A0 END UNTIL (JJ=(JJ+1))>JNDMAX;               564000  0199
IF JKOUNT=0 THEN GO TO L1A;                                                      565000  020A
IF JNDMAX>7 THEN GO TO LA3O;                                                     566000  0205
JNFORM=JNDMAX;                                                                   567000  0206
GO TO LA40;                                                                      568000  0207
LA3O: JNFORM=7;                                                                  569000  0208
LA4O: WRITE(PRINT(PAGE));                                                        570000  0208
      WRITE(PRINT,FL110,L15);                                                    571000  0212
WRITE(PRINT,FLA5O);                                                              572000  0216
WRITE(PRINT,FLA6O);                                                              573000  0219
GO TO SWGO2(JNFORM);                                                             574000  0223
LA6O: WRITE(PRINT,FLA85);                                                        575000  0225
GO TO L55O;                                                                      576000  0228
LA9O: WRITE(PRINT,FL49*);                                                        577000  0229
GO TO L55O;                                                                      578000  0232
```

```
L500: WRITE(PRINT,FL505);                                          579000  0233
GO TO L550;                                                        580000  0236
L510: WRITE(PRINT,FL515);                                          581000  0237
GO TO L550;                                                        582000  0240
L520: WRITE(PRINT,FL525);                                          583000  0241
GO TO L550;                                                        584000  0244
L530: WRITE(PRINT,FL535);                                          585000  0245
GO TO L550;                                                        586000  0248
L540: WRITE(PRINT,FL545);                                          587000  0249
L550: JI=I;                                                        588000  0252
DO BEGIN                                                           589000  0253
  WRITE(PRINT,FL560,LIST10);                                       590000  0253
  END UNTIL (JI=(JI+I))>JNMMAX;                                    591000  0257
WRITE(PRINT,FL580,LIST1);                                          592000  0259
IF JNOMAX≤JNFORM THEN GO TO L663;                                  593000  0263
JNFORM=JNOMAX;                                                     594000  0264
GO TO L440;                                                        595000  0265
L600: WRITE(PRINT,FL605);                                          596000  0266
GO TO L650;                                                        597000  0269
L610: WRITE(PRINT,FL615);                                          598000  0270
GO TO L650;                                                        599000  0273
L620: WRITE(PRINT,FL625);                                          600000  0274
L650: JI=I;                                                        601000  0277
DO BEGIN                                                           602000  0278
  WRITE(PRINT,FL560,LIST12);                                       603000  0278
  END UNTIL (JI=(JI+I))>JNMMAX;                                    604000  0282
WRITE(PRINT,FL580,LIST13);                                         605000  0284
L663: JJJ=1;                                                       606000  0288
DO BEGIN                                                           607000  0289
  SVNDET(JJJ)=JJJ END UNTIL (JJJ=(JJJ+I))>JNOMAX;                  608000  0289
JKA2=0;                                                            609000  0293
L670: WRITE(PRINT(PAGE));                                          610000  0294
```

```
      WRITE(PRINT,FL680));                                          611000  0297
WRITE(PRINT,FL690));                                                612000  0300
JKA1←JKA2←1;                                                        613000  0304
JKA2←JKA1←4;                                                        614000  0305
IF JKA2≤JNOMAX THEN GO TO L700;                                     615000  0306
JKA2←JNOMAX;                                                        616000  0308
L700: WRITE(PRINT,FL705,LIST14));                                   617000  0309
JJJ←2;                                                              618000  0312
DO BEGIN                                                            619000  0313
  WRITE(PRINT,FL710,LIST15));                                       620000  0313
  END UNTIL (JJJ←(JJJ+1))>JMAXN;                                    621000  0317
WRITE(PRINT,FL720,LIST16);                                          622000  0319
IF JKA2<JNOMAX THEN GO TO L670;                                     623000  0323
SVANG[1]←1;                                                         624000  0324
JJJ←1;                                                              625000  0325
DO BEGIN                                                            626000  0326
  SVANG[JJJ+1]←SVC1P4[JJJ] END UNTIL (JJJ←(JJJ+1))>JNPA;            627000  0326
L141 JJD←1;                                                         628000  0330
DO BEGIN                                                            629000  0331
  JJ4D←(JJD-1)*JNAYAD;                                              630000  0331
  J1AD←1;                                                           631000  0333
  DO BEGIN                                                          632000  033A
    JJJ←JJ4D+J1AD;                                                  633000  033A
    JNCARD←01                                                       634000  0335
    JKA2←0;                                                         635000  0336
    L7301 JKA1←JKA2←1;                                              636000  0337
    JKA2←JKA1←6;                                                    637000  0338
    JNCARD←JNCARD+1;                                                638000  0339
    WRITE(PUNCH,FL735,LIST17));                                     639000  0340
    IF JKA2=JNPA<1 THEN GO TO L730;                                 640000  0344
    JJN←1;                                                          641000  0346
    DO BEGIN                                                        642000  0347
```

```
      JKA2←0;                                                          643000  0347
      JKA1←JKA2←1;                                                     644000  0347
      JKA2←JKA1←5;                                                     645000  0349
      JNCARD←JNCARD←1;                                                 646000  0350
      WRITE(PUNCH,FL745,L1S118);                                       647000  0351
      IF JKA2≥JMAXH THEN GO TO L770;                                   648000  0355
      JKA1←JKA2←1;                                                     649000  0356
      JKA2←9;                                                          650000  0357
      JNCARD←JNCARD←1;                                                 651000  0358
      WRITE(PUNCH,FL747,L1S118);                                       652000  0359
      L770: END UNTIL (JJN←(JJN←1))>JNPA END UNTIL (JIAD←(JIAD←1))>    653000  0363
   JNAZAD END UNTIL (JJD←(JJD←1))>JNOMAX;                              654000  0367
IF JNAZAD=1 THEN GO TO L01;                                            655000  0370
JMAXR1←JMAXH←1;                                                        656000  0372
JI←1;                                                                  657000  0373
DO BEGIN                                                               658000  0374
  JK←1;                                                                659000  0374
  DO BEGIN                                                             660000  0374
    JJ←1;                                                              661000  0374
    DO BEGIN                                                           662000  0375
      JJAD←(JJ-1)×JNAZAD;                                              663000  0375
      JR1←0;                                                           664000  0377
      JIAD←1;                                                          665000  0378
      DO BEGIN                                                         666000  0378
        JJAOO←JJAD←JIAD;                                               667000  0378
        SVFLUX[JK,J1,JJ]←SVFLUX[JK,J1,JJAOO]←JR1;                      668000  0380
        JR1←SVFLUX[JK,J1,JJ];                                          669000  0385
        END UNTIL (JIAD←(JIAD←1))>JNAZAD END UNTIL (JJ←(JJ←1))>JNOMAX  670000  0387
    END UNTIL (JK←(JK←1))>JNPA END UNTIL (J1←(JI←1))>JMAXR1;           671000  0391
J1←1;                                                                  672000  0396
DO BEGIN                                                               673000  0397
  JJ←1;                                                                674000  0397
```

```
  DO BEGIN                                                                    675000    0398
    JJAD=(JJ-1)*JNAZAD;                                                       676000    0398
    JR2=0;                                                                    677000    0399
    JIAD=1;                                                                   678000    0400
    DO BEGIN                                                                  679000    0401
      JJADD=JJAD+JIAD;                                                        680000    0401
      SVTFLUX(JI,JJ)=SVTFLUX(JI,JJADD)+JR2;                                   681000    0402
      JR2=SVTFLUX(JI,JJ);                                                     682000    0406
      END UNTIL (JIAD=(JIAD+1))>JNAZAD END UNTIL ((JJ=(JJ+1))>JNDMAX END      683000    0407
  UNTIL (JI=(JI+1))>JMAXM1;                                                   684000    0412
JNAZAD=1;                                                                     685000    041A
SVCCAZA[1]=-1;                                                                686000    0415
JKOUNT=1;                                                                     687000    0416
GO TO LTR01;                                                                  688000    0417
LOI  END END;                                                                 689000    0418
                                                                    0018 IS 0419 LONG, NEXT SEG 0015
                                                                    0015 IS 0186 LONG, NEXT SEG 0006
PROCEDURE SHAVRAGE;                                                           690000    0319
BEGIN                                                                         691000    0319
INTEGER  DX1,JI,JJ,JK ,JINDX ;                                                692000    031"
                                                                    START OF SEGMENT ********** 0019
REAL JFPART,JFGROUP;                                                          693000    0000
FORMAT FL110(" ",X29,"FLUXES FOR DEVIATION GROUP",I3,","),                    694000    0000
                                                                    START OF SEGMENT ********** 0020
FL220("    ",12,X3,S1,7E11.3),                                                695000    0000
FL230(/" TOTAL  ",S1,7E11.3),                                                 696000    0000
FL120(/" COLLISIONS",X30,"DETECTOR"),                                         697000    0000
FL145(/"              01"),                                                   698000    0000
FL155(/"              01        02"),                                         699000    0000
FL165(/"              01        02        03"),                               700000    0000
FL175(/"              01        02        03        04"),                     701000    0000
FL185(/"              01        02        03        04        05"),           702000    0000
```

```
FL195(/                                                                 703000    0000
 "              01         02         03         04         05  ",      704000    0000
 "      06"),                                                            705000    0000
FL205(/                                                                 706000    0000
 "              01         02         03         04         05  ",      707000    0000
 "      06         07"),                                                708000    0000
FL265(/"               08"),                                            709000    0000
FL275(/"               08         09"),                                 710000    0000
FL285(/"               08         09         10"),                      711000    0000
FL320(/" BASE FOR RANDOM NUMBER GENERATOR IS",113),                     712000    0000
FL400(" ",X11,                                                          713000    0000
 " SCATTERED INTENSITIES VERSUS DETECTOR AND COLLISION NUMBER.",),      714000    0000
FL460(" ",X11,                                                          715000    0000
 " INTENSITY DEVIATIONS VERSUS DETECTOR AND COLLISION NUMBER.",)        716000    0000
                                                          0020 IS 0196 LONG, NEXT SEG 0019
LIST LIST1(JNDEVG);                                                     717000    0000
LIST LIST2(SVINCOL[J1],FOR DX1+1 STEP 1 UNTIL JNFORM DO SVAFLUX[JI,DX11)  718000  0005
 );                                                                     719000    0010
LIST LIST3(FOR DX1+1 STEP 1 UNTIL JNFORM DO SVSTFLUX[DX1]);             720000    0016
LIST LIST4(SVINCOL[J1],FOR DX1+8 STEP 1 UNTIL JNOMAX DO SVAFLUX[J1,DX11)  721000  0025
 );                                                                     722000    0030
LIST LIST5(FOR DX1+8 STEP 1 UNTIL JNOMAX DO SVSTFLUX[DX11);             723000    0036
LIST LIST6(J1BASE);                                                     724000    0045
LABEL L115,L125,L130,L140,L150,L160,L170,L180,L190,L200,L210,L310,       725000    0050
 L260,L270,L280,L290,L410,L0;                                           726000    0050
SWITCH SWG01+L140,L150,L160,L170,L180,L190,L200;                        727000    0050
SWITCH SWG02+L260,L270,L280;                                            728000    0057
COMMENT  SUBROUTINE AVRAGE;                                             729000    0062
JNDEVG+JNDEVG+1;                                                        730000    0062
JFPART+JNPART;                                                          731000    0064
J1NDX+0;                                                                732000    0065
JFGROUP+JNGROUP;                                                        733000    0065
```

```
JJ+1);                                                    734000  0066
DO BEGIN                                                  735000  0067
  SVSIFLUX(JJ1+0)                                         736000  0067
  JI+1);                                                  737000  0068
  DO BEGIN                                                738000  0069
    SVAFLUX(JI,JJ)+SVAFLUX(JI,JJ)/JFPART(                 739000  0069
    SVSAFLUX(JI,JJ1+SVSAFLUX(JI,JJ)+SVAFLUX(J),JJ1)       740000  0072
    SVSQFLUX(JI,JJ1+SVSQFLUX(JI,JJ1+SVAFLUX(JI,JJ)+2)     741000  0077
    SVSTFLUX(JJI+SVSTFLUX(JJ)+SVAFLUX(JI,JJ))             742000  0082
    END UNTIL (JI+(JI+1))>JNPCUL);                        743000  0085
  SVFFLUX(JJ1+SVFFLUX(JJ1+SVSTFLUX(JJ1);                  744000  0087
  SVDVFLUX(JJ1+SVDVFLUX(JJ)+SVSTFLUX(JJ)+2)               745000  0089
  END UNTIL (JJ+(JJ+1))>JNUMAX);                          746000  0092
WRITE(PRINT(PAGE)));                                      747000  0094
WRITE(PRINT,FL110,L1ST1));                                748000  0098
L115: WRITE(PRINT,FL120));                                749000  0101
IF JNUMAX>7 THEN GO TO L125;                              750000  0105
JNFORM+JNUMAX;                                            751000  0106
GO TO L130;                                               752000  0107
L125: JNFORM+7;                                           753000  0108
L130: GO TO SWGO1(JNFORM);                                754000  0108
L140: WRITE(PRINT,FL145);                                 755000  0111
GO TO L210;                                               756000  0114
L150: WRITE(PRINT,FL155));                                757000  0115
GO TO L210;                                               758000  0118
L160: WRITE(PRINT,FL165));                                759000  0119
GO TO L210;                                               760000  0122
L170: WRITE(PRINT,FL175));                                761000  0123
GO TO L210;                                               762000  0126
L180: WRITE(PRINT,FL185));                                763000  0127
GO TO L210;                                               764000  0130
L190: WRITE(PRINT,FL195));                                765000  0131
```

```
GO TO L210;                                                    766000  0134
L200: WRITE(PRINT,FL205);                                      767000  0135
L210: J1:=1;                                                   768000  0138
DO BEGIN                                                       769000  0139
  WRITE(PRINT,FL220,LIST2);                                    770000  0139
  END UNTIL (JI:=(J1:=1))>JNPCOL;                              771000  0143
WRITE(PRINT,FL230,LIST3);                                      772000  0145
IF JNOMAX≤JNFORM THEN GO TO L310;                              773000  0149
JNFORM:=JNOMAX-JNFORM;                                         774000  0150
WRITE(PRINT[PAGE]);                                            775000  0152
WRITE(PRINT,FL120);                                            776000  0155
GO TO SWGO2[JNFORM];                                           777000  0158
L260: WRITE(PRINT,FL265);                                      778000  0160
GO TO L290;                                                    779000  0164
L270: WRITE(PRINT,FL275);                                      780000  0165
GO TO L290;                                                    781000  0168
L280: WRITE(PRINT,FL285);                                      782000  0169
L290: J1:=1;                                                   783000  0172
DO BEGIN                                                       784000  0173
  WRITE(PRINT,FL220,LIST4);                                    785000  0173
  END UNTIL (JI:=(J1:=1))>JNPCOL;                              786000  0177
WRITE(PRINT,FL230,LIST5);                                      787000  0179
L310: WRITE(PRINT,FL320,LIST6);                                788000  0183
JJ:=1;                                                         789000  0187
DO BEGIN                                                       790000  0188
  J1:=1;                                                       791000  0188
  DO BEGIN                                                     792000  0189
    SVAFLUX[J1,JJ]:=0;                                         793000  0189
    END UNTIL (JI:=(J1:=1))>JNPCOL END UNTIL (JJ:=(JJ:=1))>JNOMAX;  794000  0191
IF JNHIST<JNMMAX THEN GO TO L0;                                795000  0195
IF (XPR:=(J1NDX))>0 THEN GO TO L0 ELSE IF XPR<0 THEN GO TO L410;  796000  0197
J1NDX:=-1;                                                     797000  0200
```

```
JJ+1)                                                                  798000    0201
DO BEGIN                                                               799000    0201
  JI+1)                                                                800000    0201
  DO BEGIN                                                             801000    0202
    SVAFLUX[JI,JJ]+SVSAFLUX[JI,JJ]/JFGROUP)                            802000    0202
    END UNTIL (JI+(JI+11)>JNPCOL)                                      803000    0206
  SVSTFLUX[JJ1+SVFFLUX[JJ]/JFGROUP)                                    804000    0208
  END UNTIL (JJ+(JJ+11)>JNUMAX)                                        805000    0210
WRITE(PRINT[PAGE])                                                     806000    0212
WRITE(PRINT,FL400))                                                    807000    0215
GO TO 1115)                                                            808000    0219
L410: JINDX+1)                                                         809000    0219
JJ+1)                                                                  810000    0220
DO BEGIN                                                               811000    0221
  J1+1)                                                                812000    0221
  DO BEGIN                                                             813000    0222
    SVAFLUX[J1,JJ]+SQRT((SVS    UX[J1,JJ]/JFGROUP+2)+(SVSAFLUX[JI,     814000    0222
     JJ1)+2/JFGROUP+3))                                                815000    0227
    END UNTIL (J1+(J1+111>JNPCOL)                                      816000    0230
  SVSTFLUX[JJ1+SQRT((SVDVFLUX[JJ]/JFGROUP+21-(SVFFLUX[JJ]+2/JFGROUP+3]1)  817000    0232
  END UNTIL (JJ+(JJ+1))>JNUMAX)                                        818000    0238
WRITE(PRINT[PAGE]))                                                    819000    0241
WRITE(PRINT,FL460))                                                    820000    0244
GO TO 1115)                                                            821000    0247
LOI  END)                                                              822000    0251
                                                                  0019 IS 0256 LONG, NEXT SEG 000A
PROCEDURE SRANGLE)                                                     823000    0319
BEGIN                                                                  824000    0319
INTEGER  JJ,J1 )                                                       825000    0319
                                                                  START OF SEGMENT ********** 0021
FORMAT FL15(/" NO ANGLE PROBABILITY COULD BE FOUND GREATER THAN",E10.3),   826000    0000
                                                                  START OF SEGMENT ********** 0022
```

```
FL34I/* INCORRECT SUBSCRIPT FOR ANGLE PROBABILITY.*);                    827000    0000
                                                                  0026 IS 0026 LONG, NEXT SEG 0021
LIST LIST1(JRN));                                                 828000    0000
LABEL L50,L20,L35,L45,L40;                                        829000    0005
COMMENT  SUBROUTINE ANGLE;                                        830000    0005
JI=1;                                                             831000    0005
DO BEGIN                                                          832000    0006
  SRRANDA(JI,BASS,JRN));                                          833000    0006
  JJ=1;                                                           834000    0007
  DO BEGIN                                                        835000    0008
    IF SVPAG(JJ)≥JRN THEN GO TO L20;                              836000    0008
    END UNTIL (JJ=(JJ+1))>JNAG;                                   837000    0009
  WRITE(PRINT,FL15,LIST1));                                       838000    0011
  JWHDA=JWHDA+1;                                                  839000    0015
  GO TO L50;                                                      840000    0016
  L20: IF JJ>1 THEN GO TO L35;                                    841000    0017
  WRITE(PRINT,FL34I);                                             842000    001V
  JWHDA=JWHDA+1;                                                  843000    0022
  GO TO L50;                                                      844000    0024
  L35: SRRANDA(JI,BASS,JRN));                                     845000    0024
  SVSANG(JI)=SVCANG(JJ-1)+(JRN*(SVCANG(JJ-1)-SVCANG(JJ)));        846000    0026
  IF IXPR=(JRN<OP)>0 THEN GO TO L40 ELSE IF XPR<0 THEN GO TO L45; 847000    0030
  JPJMI=SVPAG(JJ-1);                                              848000    0033
  SVWEIGHT(JI)=(1/(SVPAG(JJ)-JPJMI))*(SVCANG(JJ-1)-SVCANG(JJ))/(SVCANG(1
)-SVCANG(JNAG));                                                  849000    0035
                                                                  850000    0039
  GO TO L50;                                                      851000    0041
  L40: SVWEIGHT(JI)=SVWAG(JJ);                                    852000    0041
  GO TO L50;                                                      853000    0043
  L45: SVWEIGHT(JI)=1;                                            854000    0044
  L50: END UNTIL (JI=(JI+1))>JNPART;                              855000    0045
 END;                                                             856000    0048
                                                                  0021 IS 0051 LONG, NEXT SEG 0006
```

```
PROCEDURE SKRAT(ML);                                                    857000   0319
BEGIN                                                                   858000   0319
  INTEGER JJ;      REAL ADJUST;                                         859000   0319
                                                    START OF SEGMENT ********** 0023
COMMENT  THE FOLLOWING PROCEDURES ARE USED:  SHRANDA;                   860000   0000
FORMAT FL130("  LOC =",I4,"  J =",I4,"  JHR =",I4,"  JHT =",I4,"  HN =",  861000   0000
                                                    START OF SEGMENT ********** 0024
S1,E10.3/"  HMU =",S1,E10.3,"  COTH =",S1,F10.3,"  TAUH1 =",S1,E10.3,      862000   0000
"  TAUH2 =",S1,F10.3/"  PL =",S1,E10.3,"  H2 =",S1,E10.3);                 863000   0000
                                                    0024 IS 0041 LONG, NEXT SEG 0023
LIST LIST1(JLOC,JJ,JJHR,JJHT,JHN,JHH0,JCOTH,JTAUH1,JTAUH2,JPL,JH2);     864000   0000
LABEL L20,L30,L50,L58,L105,L70,L100,L110,L0;                            865000   0018
SHRANDA(J1HAS2,JHN);                                                    866000   0018
JLOC:=55;                                                               867000   0019
JPL:=0;                                                                 868000   0020
IF ABS(JCOTH)<JSHVAL THEN GO TO L20;                                    869000   0021
IF JCOTH>0 THEN GO TO L30;                                              870000   0022
L20: JHH0:=-LN(JHN);                                                    871000   0023
GO TO L50;                                                              872000   0025
L30: JUMLMIT:=(SVTAU[JHH0]-JTAUH1)/JCOTH;                               873000   0028
JADJUST:=1-EXP(-JUMLMIT);                                               874000   0030
JHH0:=-LN(1-JHN*JADJUST);                                               875000   0032
JH1T:=JH1T+JADJUST;                                                     876000   0035
L50: JTAUH2:=JTAUH1+JHH0*JCOTH;                                         877000   0036
IF JTAUH2>0 THEN GO TO L58;                                             878000   0038
JTAUH2:=0;                                                              879000   0040
JJHH:=1;                                                                880000   0040
JJHT:=2;                                                                881000   0041
JH2:=-JDLONG;                                                           882000   0042
GO TO L105;                                                             883000   0043
L58: JJ:=1;                                                             884000   0046
OD REGEN                                                                885000   0046
```

```
  IF JTAUH2<SVTAU[JJ] THEN GO TO L70;                                  886000   0086
  END UNTIL (JJ+(JJ+1))>JNUM;                                          887000   0088
JJMB+JNUM-1;                                                           888000   0090
JJMT+JNUM;                                                             889000   0091
JM2+JDLUNG;                                                            890000   0092
GO TO L105;                                                            891000   0093
L70: JJMB+JJ-1;                                                        892000   0093
JJMT+JJ;                                                               893000   0095
IF ABS(JCOTH)>JSMVAL THEN GO TO L100;                                  894000   0096
JM2+JM1;                                                               895000   0097
JPL+JRHO/((SVTAU[JJMT]-SVTAU[JJMB])/(SVMV[JJMT]-SVMV[JJMB]));          896000   0098
GO TO L110;                                                            897000   0042
L100: JM2+SVMV[JJMB]+(SVMV[JJMT]-SVMV[JJMB])*(JTAUH2-SVTAU[JJMB])/(SVTAU  898000   0042
[JJMT]-SVTAU[JJMB]));                                                  899000   0066
L105: JPL+(JM2-JM1)/JCOTH;                                             900000   0068
L110: IF JIDUMPSO THEN GO TO L0;                                       901000   0070
WRITE(PRINT,FL130,LIST1);                                              902000   0072
L0:  END;                                                              903000   0076
                                                  5023 IS 0079 LONG, NEXT SEG 0006
PROCEDURE SRINITAL;                                                    904000   0319
BEGIN                                                                  905000   0319
COMMENT  SUBROUTINE INITAL;                                            906000   0319
INTEGER JJ,JI,JK,JN ;                                                  907000   0319
                                                  START OF SEGMENT ********** 0025
JJ+1;                                                                  908000   0000
DO BEGIN                                                               909000   0000
  JLR+JNPCOL+1;                                                        910000   0000
  JI+1;                                                                911000   0002
  DO BEGIN                                                             912000   0002
    SVSAFLUX[JI,JJ]+0;                                                 913000   0002
    SVSWFLUX[JI,JJ]+0;                                                 914000   0004
    END UNTIL (JI+(JI+1))>JLR;                                         915000   0006
```

```
  JK:=1;                                                    916000   0009
  DO BEGIN                                                  917000   0009
    SVRUN[JK,JJ1:=0 END UNTIL (JK:=(JK+1))>JMAXR;           918000   0009
  JN:=1;                                                    919000   0014
  DO BEGIN                                                  920000   0014
    SVFLUO[JN,JJ1:=0;                                       921000   0014
    END UNTIL (JN:=(JN+1))>JNHMAX;                          922000   0016
  SVRFLUX[JJ]:=0;                                           923000   0019
  SVFFLUX[JJ]:=0;                                           924000   0020
  SVOVFLUX[JJ]:=0;                                          925000   0021
  SVFLUX[JJ]:=0;                                            926000   0022
  END UNTIL (JJ:=(JJ+1))>JNOMAX;                            927000   0024
JMAXP1:=JMAXM+1;                                            928000   0026
JJAMAX:=JNOMAX*JNAZA;                                       929000   0027
JJ:=1;                                                      930000   0028
DO BEGIN                                                    931000   0029
  JI:=1;                                                    932000   0029
  DO BEGIN                                                  933000   0030
    JK:=1;                                                  934000   0030
    DO BEGIN                                                935000   0031
      SVTFLUX[JJ,JI]:=0;                                    936000   0031
      SVFLUX[JP,JJ,JI]:=0 END UNTIL (JK:=(JK+1))>JNP4 END UNTIL (JI:=(  937000   0033
   JI:=1))>JJAMAX END UNTIL (JJ:=(JJ+1))>JMAXRI;            938000   0038
JMAXCOL:=INT(0);                                            939000   0042
JNMAIT:=INT(0);                                             940000   0043
JNMSTOP:=0;                                                 941000   0045
JNMAXR:=0;                                                  942000   0045
JI:=1;                                                      943000   0046
DO BEGIN                                                    944000   0047
  SVNRTCO[JI]:=INT(0) END UNTIL (JI:=(JI+1))>JNHMAX;        945000   0047
END;                                                        946000   0051
                                                0025 IS 0054 LONG, NEXT SEG 0006
```

```
PROCEDURE SRREFLCT;                                          947000    0319
BEGIN                                                        948000    0319
REAL  JDENOM;       INTEGER   JI,JJAIL;                      949000    0319
                                                  START OF SEGMENT ********** 0026
COMMENT  THE FOLLOWING PROCEDURES ARE USED:  SRRANDA;        950000    0000
FORMAT FL35(/" REFLECTION ANGLE DISTRIBUTION FOR BOUNDARY",I3,  951000    0000
                                                  START OF SEGMENT ********** 0027
 " IS IN ERROR.");                                           952000    0000
                                                  0027 IS 0016 LONG, NEXT SEG 0028
LIST LIST1(JNRR);                                            953000    0000
LABEL L10,L20,L15,L70,L50,L60,LA0,L0;                        954000    0005
SWITCH SWG01:=L10,L20,L15,L40;                               955000    0005
COMMENT  SUBROUTINE REFLCT;                                  956000    0011
SRRANDA(J1RAS5,JRN);                                         957000    0011
JJAIL:=SVJREFLT[JNRR];                                       958000    0013
GO TO SWG01[JJAIL];                                          959000    0014
L10: JCOTH1:=JRN;                                            960000    0016
GO TO L70;                                                   961000    0016
L15: JCOTH1:=-JRN;                                           962000    0017
GO TO L70;                                                   963000    0019
L20: JFNRA:=SVNREFCOS[JNRR];                                 964000    0019
JPR1:=JRN*JFNRA;                                             965000    0021
J1:=INT(JPR1);                                               966000    0022
IF (XPR*(J1))>0 THEN GO TO L60 ELSE IF XPR=0 THEN GO TO L50; 967000    0023
WRITE(PRINT,FL35,LIST1);                                     968000    0026
JNRRA:=JNRRA+1;                                              969000    0030
GO TO L0;                                                    970000    0031
L50: IF (XPR*(JJAIL-2))<0 THEN JCOTH1:=1*JPR1*(SVREFLCOS[1,JNRR]-1) ELSE     971000    0032
 JCOTH1:=JPR1*SVREFLCOS[1,JNRR];                             971100    0037
GO TO L70;                                                   972000    00A0
L60: JF1:=J1;                                                973000    00A0
JCOTH1:=SVREFLCOS[J1,JNRR]+(JPR1-JF1)*(SVREFLCOS[J1+1,JNRR]-SVREFLCOS[     974000    0041
```

```
 J1,JNRBI)!                                                    975000    0045
L70: JSITH1*SQRT(1-JCOTH1**2)!                                 976000    0048
L80: SRRANDA(JIPASE,JRN)!                                      977000    0050
JSPT*2=JRN-1!                                                  978000    0052
SRRANDA(JIPAS1,JRN)!                                           979000    0053
JCPT*2=JRN-1!                                                  980000    0054
JDENOM=JCPT**2+JSPT**2!                                        981000    0056
IF JDENOM>1 THEN GO TO L80!                                    982000    0058
JDENOM=SQRT(JDENOM)!                                           983000    0060
JCPH11=JCPT/JDENOM!                                            984000    0061
JSPH11=JSPT/JDENOM!                                            985000    0062
JCAPH1=JCPH11!                                                 986000    0064
JSAPH1=JSPH11!                                                 987000    0064
L01  END!                                                      988000    0065
                                                        0026 IS 0071 LONG, NEXT SEG 0006
PROCEDURE SRSCTANG!                                            989000    0319
BEGIN                                                          990000    0319
REAL  JCOPH1, JSOPH11   INTEGER  J1,JNPASE !                   991000    0319
                                                        START OF SEGMENT ********** 0028
COMMENT  THE FOLLOWING PROCEDURES ARE USED!  SRREFLCT,SRRANDA!  992000    0000
FORMAT FL100(/" THE PHASE ANGLE PROBABILITIES FOR MATERIAL",I3,  993000    0000
                                                        START OF SEGMENT ********** 0029
 " ARE INCORRECT,"),                                           994000    0000
FL139(/" LOC =",I4,"  NPHASE =",I4,"  NCM =",I4,"  REFL =",S1,E10.3,  995000    0000
 "  CSANG =",S1,E10.3/" SSANG =",S1,F10.3,"  CTEP =",S1,E10.3,  996000    0000
 "  STEP =",S1,E10.3,"  DEUM =",S1,F10.3,"  COPH1 =",S1,E10.3/  997000    0000
 " SOPH1 =",S1,E10.3,"  COTH2 =",S1,F10.3,"  SITH2 =",S1,E10.3,  998000    0000
 "  SOEPH1 =",S1,E10.3/" COEPH1 =",S1,E10.3,"  CPH12 =",S1,E10.3,  999000    0000
 "  SPH12 =",S1,F10.3,"  COTH1 =",S1,E10.3/" SITH1 =",S1,E10.3,  1000000    0000
 "  CPH11 =",S1,E10.3,"  SPH11 =",S1,F10.3,"  RN =",S1,E10.3,  1001000    0000
 "  CAPH1 =",S1,F10.3,"  SAPH1 =",S1,E10.3)!                    1002000    0000
                                                        0029 IS 0115 LONG, NEXT SEG 0028
```

```
LIST LIST1(JNCM)$                                                    1003000  0000
LIST LIST2(JLOC,JNPHASE,JNCM,JREFL,JCSANG,JSSANG,JC1EP,JSTEP,JOLDM,  1004000  0005
 JCDPHI,JSDPHI,JCDTH2,JSITH2,JSDEPHI,JCDEPHI,JCPHI2,JSPHI2,JCDTH1,   1005000  0017
 JSITH1,JCPH11,JSPHI1,JRN,JCAPH1,JSAPH1)$                            1006000  0029
LABEL L5,L137,L10,L50,L120,L100,L110,L130,L136,L0,L150$               1007000  0039
COMMENT  SUBROUTINE SCTANG$                                          1008000  0039
IF JREFL<0 THEN GO TO L5$                                            1009000  0039
SRREFLCT$                                                            1010000  0040
GO TO L137$                                                          1011000  0041
L5: SRRANDA(J1RAS3,JRN)$                                             1012000  0041
IF JRN>JRATLEE THEN GO TO L50$                                       1013000  0043
L10: SRRANDA(J1RASA,JRN)$                                            1014000  0044
JCSANG+1=2*JRN$                                                      1015000  0046
SRRANDA(J1RAS5,JRN)$                                                 1016000  0047
IF JRN<.5 THEN GO TO L120$                                           1017000  0048
SRRANDA(J1RASE,JRN)$                                                 1018000  0050
IF JRN<JCSANG*JCSANG THEN GO TO L120 ELSE GO TO L10$                 1019000  0051
L50: SRRANDA(J1RAS1,JRN)$                                            1020000  0053
JFNPA+SVNPHANG(JNCM)$                                                1021000  0055
JPH1+JRN*JFNPA$                                                      1022000  0056
J1+INT(JPR1)$                                                        1023000  0057
IF (XPR+(J1))>0 THEN GO TO L110 ELSE IF XPR=0 THEN GO TO L100$       1024000  0058
WRITE(PRINT,FLRO,L1ST1)$                                             1025000  0061
JWRDA+JWRDA+1$                                                       1026000  0063
GO TO L0$                                                            1027000  0066
L100: JCSANG+1+JPR1*(SVPHANG(J1+1,JNCM)-1)$                          1028000  0068
GO TO L120$                                                          1029000  0071
L110: JF1+J1$                                                        1030000  0071
JCSANG+SVPHANG(J1,JNCM)+(JPR1-JF1)*(SVPHANG(J1+1,JNCM)-SVPHANG(J1,   1031000  0072
 JNCM))$                                                             1032000  0077
L120: JSSANG+SQRT(1-JCSANG*JCSANG)$                                  1033000  0079
L130: SRRANDA(J1RAS2,JRN)$                                           103A000  0081
```

```
JCTEP*1=2*JRN)                                          1035000   0083
SHRANDA(J1RAS3,JRN))                                    1036000   008A
JSTEP*1=2*JRN)                                          1037000   0085
JDEDM*JCTEP*2+JSTEP*2)                                  1038000   0087
IF JDEDM>1 THEN GO TO L130)                             1039000   0089
JDEDM=SQRT(JDEDM))                                      1040000   0091
JCDPH1=JCTEP/JDEDM)                                     1041000   0092
JSDPH1=JSTEP/JDEDM)                                     10A2000   0093
IF JSITH2>JSMVAL THEN GO TO L136)                       10A3000   0095
JCDTH1=JCSANG*JCDTH2)                                   10AA000   0096
JSITH1=JSSANG)                                          1045000   0097
JCPH11=JCDPH1)                                          1046000   0098
JSPH11=JSDPH1)                                          10A7000   0099
    JCDEPH1=JCDPH1)                                     01047100  0099
    JSDEPH1=JSDPH1)                                     010A7200  0100
    GO TO L150)                                         01048000  0101
L136) JCDTH1=JCDTH2*JCSANG+JSITH2*JSSANG*JCDPH1)        1049000   0104
JSITH1=SQRT(1-JCDTH1**2))                               1050000   0106
JSDEPH1=(JSSANG*JSDPH1)/JSITH1)                         1051000   0109
JCDEPH1=(JCSANG-JCDTH2*JCDTH1)/(JSITH2*JSITH1))         1052000   0111
JCPH11=JCPH12*JCDEPH1-JSPH12*JSDEPH1)                   1053000   0113
JSPH11=JSPH12*JCDEPH1+JCPH12*JSDEPH1)                   1054000   0116
    L150)                                               01054500  0118
JCAPH11=JCAPH1)                                         1055000   0119
JSAPH11=JSAPH1)                                         1056000   0119
JCAPH1=JCAPH11*JCDEPH1-JSAPH11*JSDEPH1)                 1057000   0120
JSAPH1=JSAPH11*JCDEPH1+JCAPH11*JSDEPH1)                 1058000   0122
L137) JCDTH2=JCDTH1)                                    1059000   0125
JSITH2=JSITH1)                                          1060000   0125
JCPH12=JCPH11)                                          1061000   0126
JSPH12=JSPH11)                                          1062000   0127
JLOC=80)                                                1063000   0128
```

```
IF J10UMPS0 THEN GO TO L0;                                    1064000   0128
WRITE(PRINT,FL139,LIST2);                                     1065000   0130
L0:  END;                                                     1066000   0133
                                                        0026 IS 0139 LONG, NEXT SEG 0004
PROCEDURE SHOBEAM;                                            1067000   0319
BEGIN                                                         1068000   0319
INTEGER JJ, JJ2;  REAL   JVO;                                 1069000   0319
                                                        START OF SEGMENT ********** 0030
FORMAT FL11(" HS IS GREATER THAN HV(NOM),                ");  1070000   0000
                                                        START OF SEGMENT ********** 0031
FL230(" RADIATION RESEARCH ASSOCIATES -LITE- PROBLEM",I10),   1071000   0000
FL240(/" DIRECT BEAM LIGHT INTENSITIES"//                     1072000   0000
 "  DETECTOR   DIRECT INTENSITY"),                            1073000   0000
FL250(/" ",16,X8,S1,E11.3);                                   1074000   0000
                                                        0031 IS 0049 LONG, NEXT SEG 0030
LIST LIST1(JNPROB);                                           1075000   0000
LIST LIST2(JJ,SVDBFLUX[JJ]);                                  1076000   0005
LABEL L3,L100,L210,L0;                                        1077000   0012
COMMENT  SUBROUTINE BREAM;                                    1078000   0012
JJ2=2;                                                        1079000   0012
DO BEGIN                                                      1080000   0013
  IF JHS≤SVHV[JJ2] THEN GO TO L3;                             1081000   0013
  END UNTIL (JJ2=(JJ2+1))>JNUM;                               1082000   0014
WRITE(PRINT,FL11);                                            1083000   0017
GO TO L0;                                                     1084000   0020
L3: JJMB=JJ2-1;                                               1085000   0021
JJMT=JJ2;                                                     1086000   0022
JJ=1;                                                         1087000   0023
DO BEGIN                                                      1088000   0023
  JVO=SVHO[JJ]-JHS;                                           1089000   0023
  JT=SQRT(JVO*2+SVRO[JJ+2]);                                  1090000   0025
  JCOTK=JVO/JT;                                               1091000   0026
```

```
  IF ABS(JCOTH)>JSMVAL THEN GO TO L100;                          1092000   0029
  JMMJT=JT*(SVTAU1JJMT]-SVTAU1JJMB1))/(SVMV1JJMT]-SVMV1JJMB1));  1093000   0031
  GO TO L210;                                                    1094000   0035
  L100: JRMOT=(SVTAUMD1JJ]*JTAUM)/JCOTH;                         1095000   0037
  L210: SVORFLUX[JJ]=SVORSS[JJ]*EXP(-JRMOT)/JT**2;               1096000   0039
  END UNTIL (JJ=(JJ+1))>JNUMAX;                                  1097000   00A3
WRITE(PRINT[PAGE 1]);                                            1098000   00A5
WRITE(PRINT,FL230,L1ST1);                                        1099000   0048
WRITE(PRINT,FL240);                                              1100000   0052
JJ=1;                                                            1101000   0055
DO BEGIN                                                         1102000   0056
  WRITE(PRINT,FL230,L1ST2);                                      1103000   0056
  END UNTIL (JJ=(JJ+1))>JNUMAX;                                  1104000   0060
WRITE(PRINT[PAGE 1]);                                            1105000   0062
JMMDA=JMMDA+1;                                                   1106000   0065
LOC  END;                                                        1107000   0067
                                                                 0030 IS 0071 LONG, NEXT SEG 00CA
PROCEDURE SRCHECK;                                               1108000   0319
BEGIN                                                            1109000   0319
INTEGER   JI1,JINAG,JINPA,JINPCOL,JINRE1,JINRE2,JINRF,JJCHECM,JJCHECK,  1110000   0319
                                                                 START OF SEGMENT ********** 0032
        JJ,JNRE1,JNRE2,JNR3,JNRF,JNAG1,JNPA1,JNPCOL1 ;           1111000   0000
FORMAT FL25("  THE NUMBER OF REFLECTION BOUNDARIES",I3,          1112000   0000
                                                                 START OF SEGMENT ********** 0033
 " EXCEEDS THE LIMIT OF 5 ALLOWED",".DATA CHECK CONTINUES,..."),  1113000   0000
FL45("  THE NUMBER OF DETECTIONS",I3," EXCEEDS THE LIMIT OF 10 ALLOWED",  1114000   0000
 ".DATA CHECK CONTINUES,..."),                                   1115000   0000
FL65("  THE NUMBER OF MATERIALS",I3," EXCEEDS THE LIMIT OF 10 ALLOWED",  1116000   0000
 ".DATA CHECK CONTINUES,..."),                                   1117000   0000
FL85("  THE NUMBER OF PHINT COLLISIONS",I3,                      1118000   0000
 " EXCEEDS THE LIMIT OF 20 ALLOWED",".DATA CHECK CONTINUES,..."),  1119000   0000
FL105("  THE NUMBER OF PRINT ANGLES",I3,                         1120000   0000
```

```
 " EXCEEDS THE LIMIT OF 25 ALLOWED",".DATA CHECK CONTINUES..."),                1121000    0000
FL125(" THE NUMBER OF SOURCE ANGLES",I3,                                        1122000    0000
 " EXCEEDS THE LIMIT OF 37 ALLOWED",".DATA CHECK CONTINUES..."),                1123000    0000
FL145(" THE NUMBER OF REGIONS",I4," EXCEEDS THE LIMIT OF 100 ALLOWED",         1124000    0000
 ".DATA CHECK CONTINUES..."),                                                   1125000    0000
FL165(" THE NUMBER OF BOUNDARIES",I4,                                           1126000    0000
 " EXCEEDS THE LIMIT OF 100 ALLOWED",".DATA CHECK CONTINUES..."),               1127000    0000
FL180(" COSINE SOURCE ANGLES MUST BE INPUT IN DESCENDING ORDER",                1128000    0000
 " DATA CHECK CONTINUES..."),                                                   1129000    0000
FL215(" COSINE PRINT ANGLES MUST BE INPUT IN DESCENDING ORDER",                 1130000    0000
 ".DATA CHECK CONTINUES..."),                                                   1131000    0000
FL235(" REFLECTION ANGLES MUST BE INPUT IN DESCENDING ORDER",                   1132000    0000
 ".DATA CHECK CONTINUES..."),                                                   1133000    0000
FL270(" REFLECTION COSINES MUST BE INPUT IN DESCENDING ORDER",                  1134000    0000
 ".DATA CHECK CONTINUES..."),                                                   1135000    0000
FL315(" DIFFERENTIAL COSINES MUST BE INPUT IN DESCENDING ORDER",                1136000    0000
 ".DATA CHECK CONTINUES..."),                                                   1137000    0000
FL355(" PHASE ANGLES MUST BE INPUT IN DESCENDING ORDER",                        1138000    0000
 ".DATA CHECK CONTINUES..."),                                                   1139000    0000
FL385(" ANGLE PROBABILITIES MUST BE INPUT IN ASCENDING ORDER",                  11A0000    0000
 ".DATA CHECK CONTINUES..."),                                                   11A1000    0000
                                                0033 IS 0267 LONG, NEXT SEG 0032
                                                START OF SEGMENT ********** 003A
FL415(" INPUT NUMBER OF COLLISSION MUST BE IN ASCENDING ORDER",                 11A2000    0000
 ".DATA CHECK CONTINUES..."),                                                   11A3000    0000
FL435(" ","  THERE ARE A TOTAL OF",I5," INPUT DATA ERRORS"///                   11AA000    0000
 " AKE PROBLEM OFF COMPUTER AND CORRECT ERRORS, BETTER LUCK NEXT ",             11A5000    0000
 "TIME"),                                                                       11A6000    0000
FLA55(" INPUT DATA SEEMS TO BE ALLRIGHT, EXECUTION CONTINUES.");                11A7000    0000
                                                003A IS 0056 LONG, NEXT SEG 0032
LIST LIST1(JNNFLH);                                                             11A8000    0000
LIST LIST2(JNUMAX);                                                             11A9000    0005
```

```
LIST LIST3(JNMA1)$                                                    1150000  0010
LIST LIST4(JNPCOL)$                                                   1151000  0015
LIST LIST5(JNPA)$                                                     1152000  0020
LIST LIST6(JNAG)$                                                     1153000  0025
LIST LIST7(JNRMAX)$                                                   1154000  0030
LIST LIST8(JNRMAX)$                                                   1155000  0035
LIST LIST9(JJCHECK)$                                                  1156000  0040
LABEL L30,L50,L70,L90,L110,L130,L150,L170,L200,L220,L300,L240,L280,    1157000  0045
 L370,L320,L340,L390,LA20,L450$                                       1158000  0045
JJCHECK=0$                                                            1159000  0045
IF JNRFLAG$ THEN GO TO L30$                                           1160000  0046
WRITE(PRINT,FL25,LIST1)$                                              1161000  0047
JJCHECK=JJCHECK+11                                                    1162000  0051
L30$ IF JNRMAX$10 THEN GO TO L50$                                     1163000  0052
WRITE(PRINT,FLA5,LIST2)$                                              1164000  0054
JJCHECK=JJCHECK+1$                                                    1165000  0058
L50$ IF JNMAT$10 THEN GO TO L70$                                      1166000  0059
WRITE(PRINT,FLA5,LIST3)$                                              1167000  0061
JJCHECK=JJCHECK+1$                                                    1168000  0065
L70$ IF JNPCOL$2A THEN GO TO L90$                                     1169000  0066
WRITE(PRINT,FLA5,LIST4)$                                              1170000  0068
JJCHECK=JJCHECK+1$                                                    1171000  0072
L90$ IF JNPA$25 THEN GO TO L110$                                      1172000  0073
WRITE(PRINT,FL105,LIST5)$                                             1173000  0075
JJCHECK=JJCHECK+11                                                    1174000  0079
L110$ IF JNAG$37 THEN GO TO L130$                                     1175000  0080
WRITE(PRINT,FL125,LIST6)$                                             1176000  0082
JJCHECK=JJCHECK+1$                                                    1177000  0086
L130$ IF JNRMAX$100 THEN GO TO L150$                                  1178000  0087
WRITE(PRINT,FL145,LIST7)$                                             1179000  0089
JJCHECK=JJCHECK+1$                                                    1180000  0093
L150$ IF JNRMAX$100 THEN GO TO L170$                                  1181000  009A
```

```
WRITE(PRINT,FL165,LISTA);                                    1182000  0096
JJCHECK+JJCHECK+1;                                           1183000  0100
JINAG+JNAG-1;                                                1184000  0101
L170: JJ+1;                                                  1185000  0102
DO BEGIN                                                     1186000  0103
  IF SVCANG[JJ]≥SVCANG[JJ+1] THEN GO TO L200;                1187000  0103
  WRITE(PRINT,FL180);                                        1188000  0106
  JJCHECK+JJCHECK+1;                                         1189000  0109
  L200: END UNTIL (JJ+(JJ+1))>JINAG;                         1190000  0110
JINPA+JNPA-1;                                                1191000  0113
JJ+1;                                                        1192000  0114
DO BEGIN                                                     1193000  0115
  IF SVCIPA[JJ]≥SVCIPA[JJ+1] THEN GO TO L220;                1194000  0115
  WRITE(PRINT,FL215);                                        1195000  0117
  JJCHECK+JJCHECK+1;                                         1196000  0121
  L220: END UNTIL (JJ+(JJ+1))>JINPA;                         1197000  0122
IF JNPFLAG THEN GO TO L300;                                  1198000  0125
JI1+1;                                                       1199000  0126
DO BEGIN                                                     1200000  0127
  JNRF+SVNRFANG[JI1];                                        1201000  0127
  JINRF+JNRF-1;                                              1202000  0128
  JJ+1;                                                      1203000  0129
  DO BEGIN                                                   1204000  0130
    IF SVRFANG[JJ,JI1]≥SVRFANG[JJ+1,JI1] THEN GO TO L240;    1205000  0130
    WRITE(PRINT,FL235);                                      1206000  0134
    JJCHECK+JJCHECK+1;                                       1207000  0137
    L240: END UNTIL (JJ+(JJ+1))>JINRF;                       1208000  0136
  END UNTIL (JI1+(JI1+1))>JNRFLG;                            1209000  0141
JI1+1;                                                       1210000  0143
DO BEGIN                                                     1211000  0144
  JNRF1+SVNRFCOS[JI1];                                       1212000  0144
  JINRF1+JNRF1-1;                                            1213000  0145
```

```
    JJ+1)                                                                1214000   0146
    DO BEGIN                                                             1215000   0147
      IF SVRFLC(US1JJ,J111)>SVM>LC(US1JJ+1,J111 THEN GO TO L280)         1216000   0147
      WRITE(PRINT,FL270))                                                1217000   0151
      JJCHECK=JJCHECK+1)                                                 1218000   0154
      L280) END UNTIL (JJ=(JJ+11)>J1NRF1)                                1219000   0155
    END UNTIL (J11=(J11+1))>JNRFLR)                                      1220000   0158
  L300) J11+1)                                                           1221000   0160
  DO BEGIN                                                               1222000   0161
    IF SVRAYLEF1(J111=1 THEN GO TO L370)                                 1223000   0161
    JNRF2=SVNOFCUS(J111)                                                 1224000   0163
    J1NRF2=JNRF2-1)                                                      1225000   0164
    JJ+1)                                                                1226000   0165
    DO BEGIN                                                             1227000   0166
      IF SVOIFCOS(JJ,J111>SVO1FCOS(JJ+1,J111 THEN GO TO L320)            1228000   0166
      WRITE(PRINT,FL315))                                                1229000   0170
      JJCHECK=JJCHECK+1)                                                 1230000   0173
      L320) END UNTIL (JJ=(JJ+1))>J1NRF2)                                1231000   0174
    JNRF3=SVNPHANG(J111-1)                                               1232000   0177
    JJ+1)                                                                1233000   0178
    DO BEGIN                                                             1234000   0179
      IF SVPHANG(JJ,J111>SVPHANG(JJ+1,J111 THEN GO TO L360)              1235000   0179
      WRITE(PRINT,FL355))                                                1236000   0183
      JJCHECK=JJCHECK+1)                                                 1237000   0186
      L360) END UNTIL (JJ=(JJ+1))>JNRF3)                                 1238000   0188
    L370) END UNTIL (J11=(J11+1))>JNMAT)                                 1239000   0190
  JJ+1)                                                                  1240000   0193
  DO BEGIN                                                               1241000   0194
    IF SVPAG(JJ)SSVPAG(JJ+1) THEN GO TO L390)                            1242000   0194
    WRITE(PRINT,FL385))                                                  1243000   0196
    JJCHECK=JJCHECK+1)                                                   1244000   0199
    L390) END UNTIL (JJ=(JJ+1))>J1NAG)                                   1245000   0201
```

```
JINPCOL+JNPCOL-1);                                                  1246000   0203
JJ+1;                                                               1247000   020A
DO BEGIN                                                            1248000   0205
  IF SVINCOL(JJ)=SVINCOL(JJ+1) THEN GO TO L420;                     1249000   0205
  WRITE(PRINT,FL415);                                               1250000   0207
  JJCHECK=JJCHECK+1;                                                1251000   0211
  L420: END UNTIL (JJ+(JJ+1))>JINPCOL;                              1252000   0212
IF JJCHECK=0 THEN GO TO L430;                                       1253000   0215
WRITE(PRINT(PAGE));                                                 1254000   0216
WRITE(PRINT,FL435,LIST9);                                           1255000   0219
GO TO FINIS;                                                        1256000   0223
L450: WRITE(PRINT,FL455);                                           1257000   0225
 END;                                                               1258000   0229
                                                                  0032 IS 0238 LONG, NEXT SEG 030A
PROCEDURE SHMAIN;                                                   1259000   0319
BEGIN                                                               1260000   0319
INTEGER JJ2, JJMIL;                                                 1261000   0319
                                                                  START OF SEGMENT ********** 0035
REAL JCHATIO, JFRACT;                                               1262000   0000
COMMENT  THE FOLLOWING PROCEDURES ARE USED:  SRINITAL,SRSEARCH,SRAVRAGE,    1263000   0000
SRANGLE,SRSATML,SROSTBO,SRHANDA,SRSCTANG,SROFTECT;                  1264000   0000
FORMAT FL11(" HS IS GREATER THAN HV(NOM),                 "),       1265000   0000
                                                                  START OF SEGMENT ********** 0036
FL5(/" CANNOT LOCATE REGION CONTAINING SOURCE PARTICLE."),          1266000   0000
FL13A(/" PROGRAM FAILED TO CALCULATE DISTANCE TO A BOUNDARY."),     1267000   0000
FL7A(/" LOC =",I4," NPART =",I4," NSP =",I4," NHIST =",I6," NCM =",  1268000   0000
 I4," NCOL =",I4/" H1 =",S1,E10.3," R1 =",S1,E10.3," COTH1 =",S1,   1269000   0000
 E10.3," SITH1 =",S1,E10.3/" CPHI1 =",S1,F10.3," SPHI1 =",S1,E10.3,  1270000   0000
 " WAIT =",S1,F10.3),                                               1271000   0000
FL9A(/" LOC =",I4," NCM =",I4," NCM =",I3," R =",S1,E10.3," H =",S1,  1272000   0000
E10.3/" COTH =",S1,F10.3," SITH =",S1,F10.3," CIPH =",S1,E10.3,     1273000   0000
 " SPHI =",S1,E10.3),                                               1274000   0000
```

```
FL10A(/" A NEGATIVE OR ZERO PATH LENGTH WAS GENERATED, PL=",S1,E10.3),          1275000  0000
FL1A2(/" LOC =",I4," NCR =",IA," NCO =",IA," T =",S1,E10.3,                     1276000  0000
 "  SUMDST =",S1,E10.3/" DIST =",S1,E10.3," RHOT =",S1,E10.3," DT =",          1277000  0000
 S1,E10.3," MT =",S1,E10.3/" NHO =",S1,E10.3," NCM =",IA," NLM =",IA)          1278000  0000
,                                                                               1279000  0000
FL18F(/" LOC =",I4," NCM =",IA," NLM =",I4," H =",S1,E10.3," TS =",             1280000  0000
 S1,E10.3/" RT =",S1,E10.3," CPHI =",S1,E10.3," R =",S1,E10.3),                 1281000  0000
FL177(/" CANNOT FIND REGION CONTAINING PARTICLE COORDINATES, H=",S1,            1282000  0000
 E10.3," R=",S1,E10.3),                                                         1283000  0000
FL284(/" LOC =",IA," NCH1 =",I4," NCH2 =",IA," DIST =",S1,E10.3,                1284000  0000
 "  DT =",S1,E10.3/" T =",S1,E10.3," SUMDST =",S1,E10.3," H2 =",S1,            1285000  0000
 E10.3," TS =",S1,E10.3/" HT =",S1,E10.3," CPH12 =",S1,E10.3," H2 =",          1286000  0000
S1,E10.3," SPH12 =",S1,E10.3/" COTH2 =",S1,E10.3," SITH2 =",S1,E10.3,          1287000  0000
 "  NCOL =",IA))                                                               1288000  0000
                                                                    0036 IS 0289 LONG, NEXT SEG 0035
LIST LIST1(JLOC,JNPART,JNS,,JNHIST,JNCR,JNCOL,JH1,JR1,JCOTH1,JSITH1,           1289000  0000
 JCPHI1,JSPHI1,JWAIT))                                                          1290000  0014
LIST LIST2(JLOC,JNCR,JNCH,JH,JM,JCOTH,JSITH,JCPHI,JSPHI))                       1291000  0020
LIST LIST3(JPL))                                                                1292000  0035
LIST LIST4(JLOC,JNCR,JNCH,JT,JSUMDST,JDIST,JRHOT,JDT,JH1,JRHO,JNCH,            1293000  0040
 JNLM))                                                                         1294000  0055
LIST LIST5(JLOC,JNCH,JNLM,JH,JTS,JRT,JCPHI,JR))                                 1295000  0059
LIST LIST6(JH,JR))                                                              1296000  0073
LIST LIST7(JLOC,JNCR1,JNCH2,JDIST,JDT,JT,JSUMDST,JH2,JTS,JRT,JCPH12,            1297000  0080
 JH2,JSPH12,JCOTH2,JSITH2,JNCOL))                                               1298000  0095
BEGIN                                                                           1299000  0104
LABEL L3,L2,L8,L7,L10,L60,L70,L3A0,L80,L100,L110,L130,L1A0,L1A4,L250,            1300000  0104
                                                                    START OF SEGMENT ********** 0037
 L550,L600,L150,L170,L1600,L165,L161,L166,L260,L180,L188,L310,L257,             1301000  0000
 L25A,L269,L320,L0,L1800)                                                       1302000  0000
SWITCH SWG01=L105,L165,L161,L161)                                               1303000  0000
COMMENT  SUBROUTINE MAIN(PLANE))                                                130A000  0005
```

```
JNPART←JNHMAX DIV JNGROUP;                                          1305000  0005
JNSP←JNPART+1;                                                      1306000  0007
JNHIST←0;                                                           1307000  0008
JNUEVG←0;                                                           1308000  0009
SHINITAL;                                                           1309000  0010
JNPREG←JNSOREG;                                                     1310000  0010
JHHOA←0;                                                            1311000  0011
JH←JHS;                                                             1312000  0012
JR←0;                                                               1313000  0012
JJ2←2;                                                              1314000  0013
DO BEGIN                                                            1315000  0014
  IF (XPR←(JHS-SVHV[JJ2]))<0 THEN GO TO L3 ELSE IF XPR=0 THEN GO TO 1316000  0014
   L2;                                                              1317000  0017
  END UNTIL (JJ2←(JJ2+1))>JNOH;                                     1318000  0018
WRITE(PRINT,FL11);                                                  1319000  0020
GO TO L0;                                                           1320000  0023
L3: JTAUH←SVTAU[JJ2-1]+(SVTAU[JJ2]-SVTAU[JJ2-1])*(JHS-SVHV[JJ2-1])/(SVHV 1321000  0024
[JJ2]-SVHV[JJ2-1]);                                                 1322000  0029
GO TO L8;                                                           1323000  0032
L2: JTAUH←SVTAU[JJ2];                                               1324000  0032
L8: JERHORS←JHHOA;                                                  1325000  0034
SRSEARCH;                                                           1326000  0034
IF JERRORS<JHHOA THEN GO TO L0;                                     1327000  0035
IF JNCR=JNSOREG THEN GO TO L7;                                      1328000  0036
WRITE(PRINT,FL6);                                                   1329000  0037
GO TO L0;                                                           1330000  0041
L7: JREFL←0;                                                        1331000  0041
L10: IF (XPH←(JNPART-JNSP))>0 THEN GO TO L70 ELSE IF XPH<0 THEN GO TO 1332000  0042
 L60;                                                               1333000  0046
SRAVRAGE;                                                           1334000  0046
IF JNHIST<JNHMAX THEN GO TO L60;                                    1335000  0047
GO TO L0;                                                           1336000  0048
```

```
L40: SRANGL(5)                                                  1337000  0048
IF JERRORS<JNMDA THEN GO TO L340;                               1338000  0049
JNSP=0;                                                         1339000  0050
L70: JNMIST=JNMIST+1;                                           1340000  0051
JNMEFL=1;                                                       1341000  0053
JLOC=10;                                                        1342000  0054
JNSP=JNSP+1;                                                    1343000  0054
JM1=0;                                                          1344000  0056
JTAUM2=JTAUM1;                                                  1345000  0056
JM1=JM5;                                                        1346000  0057
JNCM=JNSOREG;                                                   1347000  0058
JCOTM1=SVSANG(JNSP);                                            1348000  0059
JSITM1=SQRT(1-JCOTM1*JCOTM1);                                   1349000  0060
JCPM1=1;                                                        1350000  0062
JSPM1=0;                                                        1351000  0063
JWAIT=SVWEIGHT(JNSP);                                           1352000  0064
JCAPM=1;                                                        1353000  0065
JSAPM=0;                                                        1354000  0065
JNLOL=1;                                                        1355000  0066
IF JIDUMPSO THEN GO TO L80;                                     1356000  0067
WRITE(PRINT,FL76,LIST1);                                        1357000  0068
L80: JLOC=20;                                                   1358000  0072
JM=JR1;                                                         1359000  0073
JM=JM1;                                                         1360000  0074
JMEFL=0;                                                        1361000  0075
JTAUM1=JTAUM2;                                                  1362000  0075
JCOTM=JCOTM1;                                                   1363000  0076
JSITM=JSITM1;                                                   1364000  0077
JCPM=JCPM1;                                                     1365000  0078
JSPM=JSPM1;                                                     1366000  0079
JNCM=JNCR;                                                      1367000  0079
JNCM=SVMAT(JNCM);                                               1368000  0080
```

```
IF JIDUMPSO THEN GO TO L100;                                    1369000  0081
 WRITE(PRINT,FL96,LIST2);                                       1370000  0082
L100: SKPATH;                                                   1371000  0086
IF JERRORS<JMMDA THEN GO TO L340;                               1372000  0087
IF JPL>0 THEN GO TO L110;                                       1373000  0088
WRITE(PRINT,FL10A,LIST3);                                       1374000  0090
JMMDA=JMMDA+1;                                                  1375000  0093
GO TO L340;                                                     1376000  0095
L110: JT=JPL;                                                   1377000  0097
JMMDT=0;                                                        1378000  0097
JDI=0;                                                          1379000  0098
JSUMDST=0;                                                      1380000  0099
JMT=JM;                                                         1381000  0100
L130: SKDSTHD;                                                  1382000  0100
IF JERRORS<JMMDA THEN GO TO L340;                               1383000  0101
IF JNCR20 THEN GO TO L140;                                      1384000  0102
WRITE(PRINT,FL136);                                             1385000  0104
GO TO L9;                                                       1386000  0107
L1401: JSUMDST=JSUMDST+JDISTJ;                                  1387000  0108
JDIC=50;                                                        1388000  0109
IF JIDUMPSO THEN GO TO L144;                                    1389000  0110
WRITE(PRINT,FL102,LISTA);                                       1390000  0111
L144: IF JSUMDST>JT THEN GO TO L250;                            1391000  0115
JNCM=SVM=TIJNCR;                                                1392000  0116
JM=JM+JCOTH=JDISTJ;                                             1393000  0117
JTS=JDIST=JSITH;                                                1394000  0119
JRT=SQRT(JM=JR+JTS=JTS+2=JM=JTS=JCPHI);                         1395000  0120
IF JRT>JSMVAL THEN GO TO L550;                                  1396000  0125
JCPHI=1;                                                        1397000  0126
JSPHI=0;                                                        1398000  0127
GO TO L600;                                                     1399000  0128
L550: JCPHI=(JTS+JR=JCPHI)/JRT;                                 1400000  0130
```

```
JSPH1=JH*JSPH1/JR1;                                    1401000   0132
L1600: JH=JHT;                                         1402000   0134
JNLM=JNCM;                                             1403000   0134
JLUC=60;                                               1404000   0135
IF JTDUMP<0 THEN GO TO L150;                           1405000   0136
WRITE(PRINT,FL1A7,LIST);                               1406000   0137
L150: IF SVNRDUNO[JNCR]>0 THEN GO TO L170;             1407000   0141
JH2=JH-2*JDELTA*JCOTH;                                 1408000   0143
 JJ2=2;                                               01408100   0145
 DO BEGIN                                             01408200   0146
 IF (XPH=(JH2-SVHV[JJ2-1]))<0 THEN GO TO L1600;       01408300   0146
 END UNTIL (JJ2=(JJ2+1))>JNUM;                        01408400   0149
 JJ2=JNUM;                                            01408500   0151
 L1600: JTAUH2=SVTAU[JJ2-1]+(SVTAU[JJ2-1]-SVTAU[JJ2-1])*
       (JH2-SVHV[JJ2-1])/(SVHV[JJ2-1]-SVHV[JJ2-1]);   01408700   0154
JH2=JH-2*JDELTA*JSITH*JCPH1;                           1409000   0159
IF JNCR#1 THEN GO TO L1600;                            1410000   0162
JNREFL=JNREFL+1;                                       1411000   0163
IF JNREFL=JMAXR<1 THEN GO TO L1600;                    1412000   0164
JNMAXR=JNMAXR+1;                                       1413000   0166
GO TO L10;                                             1414000   0167
L1600: JREFL=1;                                        1415000   0168
JNRR=JNCR;                                             1416000   0168
JJA1L=SVJREFLT[JNRR];                                  1417000   0169
GO TO SWG01[JJA1L];                                    1418000   0170
L1A1: JCOTH2=-1;                                       1419000   0172
GO TO L166;                                            1420000   0174
L165: JCOTH2=1;                                        1421000   0174
L166: JSITH2=0;                                        1422000   0175
JCPH12=1;                                              1423000   0176
JSPH12=0;                                              1424000   0177
JWAIT=JWAIT*(SVALBFD01JNCR]+SVSIGNDT1JNCR]*JCOTH1);    1425000   0178
```

```
GO TO L260;                                                    1426000  0181
L170: JMPREG←SVMPR[JJ1,JNCH];                                  1427000  0181
SRSEARCH;                                                      1428000  0183
IF JEHRURS<JWHUA THEN GO TO L340;                              1429000  0184
IF JNCR>0 THEN GO TO L180;                                     1430000  0185
WRITE[PRINT,FL177,LIST6];                                      1431000  0186
GO TO L0;                                                      1432000  0190
L180: JNCH2←JNCH;                                              1433000  0191
IF SVEMP[JNCR2]≥SVEMP[JNCR1] THEN GO TO L188;                  1434000  0191
SHRANDA[JIBAS4,JRN];                                           1435000  0193
IF JRN>(SVEMP[JNCR2]/SVEMP[JNCR1]) THEN GO TO L310;            1436000  0194
JWAIT←JWAIT×(SVEMP[JNCR1]/SVEMP[JNCR2]);                       1437000  0196
GO TO L188;                                                    1438000  0199
L310: SVNRTCO[JNCH2]←SVNRTCO[JNCR2]+1;                         1439000  0199
JNRSTOP←JNRSTOP+1;                                             1440000  0202
GO TO L10;                                                     1441000  0203
L188: JOT←JOT+JD1ST;                                           1442000  0203
GO TO L130;                                                    1443000  0205
L250: JDIST←J1-JOT;                                            1444000  0205
JH2←JH+JCOTH×JD1ST;                                            1445000  0207
JTS←JDIST×JS1TH;                                               1446000  0209
JHT←SQRT(JH×JH+JTS×JTS+2×JH×JTS×JCPH1);                        1447000  0210
IF JRT>JSHVAL THEN GO TO L257;                                 1448000  0215
JCPHI2←1;                                                      1449000  0216
JSPHI2←0;                                                      1450000  0217
GO TO L258;                                                    1451000  0218
L257: JCPHI2←(JTS+JH×JCPH1)/JRT;                               1452000  0220
JSPHI2←JR×JSPH1/JRT;                                           1453000  0222
L258: JH2←JHT;                                                 1454000  0224
JCOTH2←JCOTH;                                                  1455000  0224
JS1TH2←JS1TH;                                                  1456000  0225
JEMACT←(JH2-SVHV[JJHR])/(SVHV[JJHT1-SVHV[JJHR]);               1457000  0226
```

```
JSHAT=0*SVSCATR(JJHR)+(SVSCATR(JJHT1)-SVSCATR(JJHB))*JFRACT)          1458000   0229
JRATLEE=SVR4YR(JJHR1+1)-SVRAYR(JJHT)-SVR4YR(JJHB))*JFRACT)            1459000   0232
JW4IT=JW4IT+JSRATIO)                                                  1460000   0235
L260: JNCR=JNCR+JNCN)                                                 1461000   0236
JLOC=70)                                                              1462000   0237
SRSCT4NG)                                                             1463000   0238
IF JERRORS<JWHOA THEN GO TO L340)                                     1464000   0239
SRDETECT)                                                             1465000   0240
IF JERRORS<JWHOA THEN GO TO L340)                                     1466000   0240
IF JTOUMPSO THEN GO TO L269)                                          1467000   0242
WRITE(PRINT,FL264,LIST7))                                             1468000   0243
L269: JNCOL=JNCOL+1)                                                  1469000   0247
      JNOGO=JNOGO+1)                                                  1469001   0248
IF JNCOL<JNCMAX THEN GO TO L320)                                      1470000   0249
JMAXCOL=JMAXCOL+1)                                                    1471000   0250
GO TO L10)                                                            1472000   0252
L320:                                                                 1473000   0252
JH1=JH2)                                                              1474000   0253
JH1=JR2)                                                              1475000   0253
JNCR=JNCR2)                                                           1476000   0254
IF JW4IT>JWCO THEN GO TO L60)                                         1477000   0255
JNW4IT=JNW4IT+1)                                                      1478000   0256
GO TO L10)                                                            1479000   0257
L340: IF JWHO4>JELIM THEN GO TO L0)                                   1480000   0258
JERRORS=JWHO4)                                                        1481000   0260
GO TO L10)                                                            1482000   0261
L0:  END END)                                                         1483000   0261
                                                        0037 IS 0263 LONG, NEXT SEG 0035
                                                        0035 IS 0110 LONG, NEXT SEG 0006
PROCEDURE SRINPUT)                                                    1484000   0319
BEGIN                                                                 1485000   0319
OWN INTEGER OX1,OX2)                                                  1486000   0319
```

```
                                                                    START OF SEGMENT ********** 0038
      INTEGER J11,J12,J13,J14,J1CHECK,JJAIL,JLIS1,JLIS2,JI,JJ                1487000    0000
      COMMENT  THE FOLLOWING PROCEDURES  ARE USED:  SHCHECK;                 1488000    0000
      FORMAT FL1(5I10),                                                      1489000    0000
                                                                    START OF SEGMENT ********** 0039
      FL170(4R10.8),                                                         1490000    0000
      FL110(2I10,4R10.8),                                                    1491000    0000
      FL130(4R10.8),                                                         1492000    0000
      FL210(2I10,R10.8),                                                     1493000    0000
      FL230(3I5,R5.2,R15),                                                   1494000    0000
      FL310(2R10.8,I10,R10.8),                                               1495000    0000
      FL410(6I10),                                                           1496000    0000
      FL510(2R10.7),                                                         1497000    0000
      FL610(6I10),                                                           1498000    0000
      FL2(X2,"PRODUCT OF NSZA AND NOMAX HAS EXCEEDED 40"/X2,                 1499000    0000
       "JOB IS TERMINATED"),                                                 1500000    0000
      FL905(/                                                                1501000    0000
       " THE NUMBER OF HISTORIES WAS NOT EQUALLY DIVISIBLE BY  THE NUMB",    1502000    0000
       "ER OF DEVIATION GROUPS."/"  THE NUMBER OF HISTORIES WAS  RESET TO",I6)1503000    0000
      ,                                                                      1504000    0000
      FL920(/" INPUT NUMBER OF MATERIALS DOES NOT AGREE WITH NMAT. "),       1505000    0000
      FL950(/" INPUT NUMBER OF BOUNDARIES DOES NOT AGREE WITH NBMAX."),      1506000    0000
      FL980(/" INPUT NUMBER OF REGIONS DOES NOT AGREE WITH NRMAX."),         1507000    0000
      FL1010(/" INPUT NUMBER OF DETECTORS DOES NOT AGREE WITH NDMAX."),      1508000    0000
      FL1040(/" INPUT NUMBER OF POINT COLLISIONS DOES NOT AGREE WITH NPCOL."),1509000   0000
      FL1070(/" INPUT NUMBER OF POINT COSINES DOES NOT AGREE WITH NPA."),    1510000    0000
      FL2000(/                                                               1511000    0000
       " INPUT NUMBER OF REFLECTION BOUNDARIES DOES NOT AGREE WITH NREFL","R.")1512000   0000
      ,                                                                      1513000    0000
      FL2030(/" INPUT SOURCE ANGLE OPTION DOES NOT AGREE WITH NAOP."),       1514000    0000
      FL2060(/" INPUT NUMBER OF SOURCE ANGLES DOES NOT AGREE WITH NAG."),    1515000    0000
      FL330(" WD(J) IS GREATER THAN WX(NOM)  FOR  J2= ",I4,"."))             1516000    0000
```

```
                                                                    0039 IS 0225 LONG, NEXT SEG 0038
LIST LIST1(JLIBMAY,J1I,JI2,JI3,JIA);                                1517000  0000
LIST LIST2(FOR DX1=1 STEP 1 UNTIL JNOH DO ISVHV(DXII,SVTAU(DXII-SVSCATH(  1518000  0010
DXII,SVHAYR(DXII);                                                  1519000  0016
LIST LIST3(SVNDFCOS(JIII,SVNPHANG(JIII,SVRAYLEE(JIII);              1520000  0024
LIST LIST4(FOR DX1=I STEP 1 UNTIL JLISI DO SVOIFCOSIDXI,JII);       1521000  0033
LIST LIST5(FOR DXI=I STEP 1 UNTIL JLIS1 DO SVPDCOS(DX1,JIII);       1522000  0043
LIST LIST6(FOR DX1=1 STEP 1 UNTIL JLIS? DO SVPHANG(DXI,JIII);       1523000  0053
LIST LIST7(FOR DX1=1 STEP 1 UNTIL JII DO (SVNBDUNO(DXII,SVITYPE(DXII,SVC  1524000  0063
OELIDX1));                                                          1525000  0069
LIST LIST8(FOR DXI=I STEP 1 UNTIL JI2 DO (SVNREGIDXII,SVNB(DXII,SVMAT(  1526000  0075
 DX1),SVEMPIDXII,FOR DX2=1 STEP I UNTIL 4 DO (SVTMIDX2,DXII,SVMPR(DX2,  1527000  0081
 DXIIII);                                                           1528000  0088
LIST LIST9(FOR DX1=1 STEP 1 UNTIL JII DO (SVMOIDXI),SVRDIDXII,SVNPMIN(  1529000  0097
 DXII,SVDRSSIDXIII);                                                1530000  0103
LIST LIST10(FOR DX1=I STEP 1 UNTIL JII DO SVIPCOL(DXII);            1531000  0111
LIST LIST11(FOR DXI=I STEP 1 UNTIL JI2 DO SVCIPA(DXII);             1532000  0120
LIST LIST12(FOR DX1=1 STEP 1 UNTIL JNAZA DO SVCAZAIDX1I);           1533000  0129
LIST LIST13(SVALREDO(JIII,SVSIGNOTIJT1));                           1534000  0138
LIST LIST1A(FOR DXI=1 STEP 1 UNTIL JI3 DO SVRFANGIDX1,JI1));        1535000  0145
LIST LIST19(FOR DXI=I STEP 1 UNTIL JI3 DO SVPOR(DXI,JIII);          1536000  0155
LIST LIST16(FOR DX)=I STEP 1 UNTIL JIA DO SVRFLCOS(DXI,JIII);       1537000  0165
LIST LIST17(FOR DX1=I STEP 1 UNTIL JI2 DO SVCANG(DX1I);             1538000  0175
LIST LIST18(FOR DXI=1 STEP 1 UNTIL JI2 DO SVPAG(DXII);              1539000  0184
LIST LIST19(FOR DXI=1 STEP 1 UNTIL JI2 DO SVWAG(DXII);              1540000  0193
LIST LIST20(JHS,JOLONG,JOELIA,JSHVAL,JWCO,JELIM,JDMIN);             1541000  0202
LIST LIST21(JNHMAX,JNGROUP,JNRMAX,JNRMAX,JNCMAX,JNOMAX,JNPA,JNPCOL,  1542000  0215
 JNAOP,JNAG,JNNFLB,JNMAT,JNSDNEG,JNAXR,JIBASF,JIBAS1,JIBAS2,JIBAS3,  1543000  0226
 JIBA$4,JIBAS5);                                                    1544000  0239
LIST LIST22(JNHMAX);                                                1545000  0244
LIST LIST23(JJ);                                                    1546000  0249
BEGIN                                                               1547000  025A
```

```
LABEL L5,L800,L700,L600,L500,L400,L300,L200,L100,L50,L900,L3000,L105,          15A8000   025A
                                                                  START OF SEGMENT ********** 0040
 L10A,L107,L1A0,L505,L506,L507,L520,L5A0,L111,L908,L930,L960,L990,             15A9000   0000
 L1020,L1050,L1080,L2010,L2040,L2070,L2087,L350,L340,L380,L01;                 1550000   0000
SWITCH SWG01:=L800,L700,L600,L500,LA00,L300,L200,L100,L50,L900,L3000;          1551000   0000
SWITCH SWG02:=L5,L520,L5,L520;                                                1552000   0008
JNMATP:=0;                                                                    1553000   001A
JNBMAXP:=0;                                                                   1554000   0015
JNMMAXP:=0;                                                                   1555000   0016
JNMFLBP:=0;                                                                   1556000   0017
JNOMAXP:=0;                                                                   1557000   0018
JNPCOLP:=0;                                                                   1558000   0018
JNPAP:=0;                                                                     1559000   0019
JNAGP:=0;                                                                     1560000   0020
L5: JNOGO:=0;                                                                 1561000   0021
READ(CARD,FL10,L1ST1)[FINIS];                                                 1562000   0021
GO TO SWG01[JLIBRAY];                                                         1563000   0026
L50: JNOM:=J11;                                                               1564000   0028
READ(CARD,FL170,L1ST2)[FINIS];                                                1565000   0029
GO TO L5;                                                                     1566000   003A
L100: JNMATP:=JNMATP+1;                                                       1567000   0038
SVMATERL[JNMATP]:=J11;                                                        1568000   0039
J1:=1;                                                                        1569000   00A0
DO BEGIN                                                                      1570000   00A1
  IF SVMATERL[J1]#SVMATERL[JNMATP] THEN GO TO L105;                           1571000   00A1
  IF J1#JNMATP THEN GO TO L106;                                               1572000   00A3
  L105: END UNTIL (J1:=(J1+1))>JNMATP;                                        1573000   004A
GO TO L107;                                                                   157A000   0047
L106: JNMATP:=JNMATP-1;                                                       1575000   00A7
L107: READ(CARD,FL110,L1ST3)[FINIS];                                          1576000   00A9
JL1S1:=SVNOFCOS[J11];                                                         1577000   0055
JL1S2:=SVNPHANG[J11];                                                         1578000   0056
```

```
IF SVHAYLEE[J11]21 THEN GO TO L5;                          1579000  0057
IF JLIS1=0 THEN GO TO L140;                                1580000  0058
READ(CARD,FL130,LIST4)[FINIS];                             1581000  0059
READ(CARD,FL130,LIST5)[FINIS];                             1582000  0064
L140: READ(CARD,FL130,LIST6)[FINIS];                       1583000  0069
GO TO L5;                                                  1584000  0075
L200: JNRMAXP=J11;                                         1585000  0080
JNRMAXP=J12;                                               1586000  0080
READ(CARD,FL210,LIST7)[FINIS];                             1587000  0081
READ(CARD,FL230,LIST8)[FINIS];                             1588000  0086
GO TO L5;                                                  1589000  0091
L300: JNRMAXP=J11;                                         1590000  0094
READ(CARD,FL310,LIST9)[FINIS];                             1591000  0094
GO TO L5;                                                  1592000  0099
L400: JNRCOLP=J11;                                         1593000  0102
JNRAP=J12;                                                 1594000  0102
READ(CARD,FL410,LIST10)[FINIS];                            1595000  0103
READ(CARD,FL130,LIST11)[FINIS];                            1596000  0108
JNAZA=J13;                                                 1597000  0113
READ(CARD,FL130,LIST12)[FINIS];                            1598000  0114
GO TO L5;                                                  1599000  0119
L500: JNRFLRP=JNRFLRP+1;                                   1600000  0123
SVJNRFLT[J11]=J12;                                         1601000  0124
SVNRFH[JNRFLRP]=J11;                                       1602000  0125
J1=1;                                                      1603000  0126
DO BEGIN                                                   1604000  0127
  IF SVNRFH[J11]=SVNRFH[JNRFLRP] THEN GO TO L505;          1605000  0127
  IF J1=JNRFLRP THEN GO TO L506;                           1606000  0129
  L505: END UNTIL (J1=(J1+1))>JNRFLRP;                     1607000  0130
GO TO L507;                                                1608000  0133
L506: JNRFLRP=JNRFLRP-1;                                   1609000  0133
L507: READ(CARD,FL510,LIST13)[FINIS];                      1610000  0135
```

```
JJA1L=SVJREFLT1JI11)                                    1611000   0141
GO TO SWGO21JJA1L1)                                     1612000   0142
L520: IF J13$0 THEN GO TO L540)                         1613000   0144
SVNRFANG1J111=J13)                                      1614000   01A5
READ(CARD,FL130,L1ST1811F1N1S1)                         1615000   0146
READ(CARD,FL130,L1ST1511F1N1S1)                         1616000   0151
L540: SVNRFCOS(J111=J14)                                1617000   0156
READ(CARD,FL130,L1ST1611F1N1S1)                         1618000   0158
GO TO L5)                                               1619000   0163
L600: JNAOPP=J11)                                       1620000   0168
JNAGP=J12)                                              1621000   0168
READ(CARD,FL130,L1ST17)(F1N1S1)                         1622000   0169
READ(CARD,FL130,L1ST1811F1N1S1)                         1623000   017A
IF JNAOPP$0 THEN GO TO L5)                              1624000   0179
READ(CARD,FL130,L1ST1911F1N1S1)                         1625000   0180
GO TO L5)                                               1626000   0185
L700: READ(CARD,FL130,L1ST20)1F1N1S1)                   1627000   0190
GO TO L5)                                               1628000   0195
L800: READ(CARD,FL810,L1ST21)1F1N1S1)                   1629000   0197
GO TO L5)                                               1630000   0202
L900: JNPROB=J11)                                       1631000   020A
J111=JNAZA=JNDMAX)                                      1632000   020A
IF J111$40 THEN GO TO L111)                             1633000   0206
WRITE(PRINT,FL21)                                       1634000   0207
WRITE(PRINT,FL2))                                       1635000   0210
ERROR(01)                                               1636000   0214
L111: J1DUMP=J12)                                       1637000   0215
J1CHECK=J13)                                            1638000   0215
JNPART=JNMMAX 01V JNGROUP)                              1639000   0216
IF JNMMAX=JNPART=JNGROUP THEN GO TO L908)               1640000   0217
JNMMAX=JNPART=JNGROUP)                                  1641000   0219
WRITE(PRINT,FL905,L1ST221)                              1642000   0220
```

```
L90A: IF JNMATP=JNMAT THEN GO TO L930;                    1643000  0224
WRITE(PRINT,FL920);                                       1644000  0226
JNUGO=JNOGO+1;                                            1645000  0229
L930: IF JNBMAXP=JNBMAX THEN GO TO L960;                  1646000  0231
WRITE(PRINT,FL950);                                       1647000  0232
JNUGO=JNOGO+1;                                            1648000  0235
L960: IF JNNMAXP=JNNMAX THEN GO TO L990;                  1649000  0237
WRITE(PRINT,FL980);                                       1650000  0238
JNUGO=JNOGO+1;                                            1651000  0241
L990: IF JNOMAXP=JNOMAX THEN GO TO L10205;                1652000  0243
WRITE(PRINT,FL1010);                                      1653000  0244
JNUGO=JNOGO+1;                                            165400   0247
L10205: IF JNPCOLP=JNPCOL THEN GO TO L10505;              1655000  0249
WRITE(PRINT,FL1040);                                      1656000  0250
JNUGO=JNOGO+1;                                            1657000  0253
L10505: IF JNPAP=JNPA THEN GO TO L10805;                  1658000  0255
WRITE(PRINT,FL1070);                                      1659000  0256
JNUGO=JNOGO+1;                                            1660000  0259
L10805: IF JNRFLBP=JNRFLB THEN GO TO L20105;              1661000  0261
WRITE(PRINT,FL2000);                                      1662000  0262
JNUGO=JNOGO+1;                                            1663000  0265
L20105: IF JNAOPP=JNAOP THEN GO TO L20405;                1664000  0267
WRITE(PRINT,FL2030);                                      1665000  0268
JNUGO=JNOGO+1;                                            1666000  0271
L20405: IF JNAGP=JNAG THEN GO TO L20705;                  1667000  0273
WRITE(PRINT,FL2060);                                      1668000  0274
JNUGO=JNOGO+1;                                            1669000  0277
L20705: IF JNOGO>0 THEN GO TO L5;                         1670000  0279
IF JTCHECKSO THEN GO TO L2087;                            1671000  0280
SNCHECK;                                                  1672000  0281
L2087: JJ1=2;                                             1673000  0282
JJ=1;                                                     1674000  0282
```

```
00 BEGIN                                                                    1675000    0283
  JJ2←JJ1;                                                                  1676000    0283
  DO BEGIN                                                                  1677000    0284
    IF (XPR←(SVHD[JJ]-SVHV[JJ2]))<0 THEN GO TO L350 ELSE IF XPR=0 THEN G    1678000    0284
0 TO L340;                                                                  1679000    0287
    END UNTIL (JJ2←(JJ2+1))>JNDH;                                           1680000    0286
  WRITE(PRINT,FL330,LIST23);                                                1681000    0290
  GO TO L3000;                                                              1682000    0294
  L350: SVTAUHD[JJ]←SVTAU[JJ2-1]+(SVTAU[JJ2]-SVTAU[JJ2-1])×(SVHD[           1683000    0294
   JJ]-SVHV[JJ2-1])/(SVHV[JJ2]-SVHV[JJ2-1]);                                1684000    0298
  GO TO L380;                                                               1685000    0303
  L340: SVTAUHD[JJ]←SVTAU[JJ2];                                             1686000    0303
  L380: JJ1←JJ2;                                                            1687000    0305
  END UNTIL (JJ←(JJ+1))>JNUMAX;                                             1688000    0306
GO TO L0;                                                                   1689000    0309
GO TO L5;                                                                   1690000    0309
L3000: ERROR(0);                                                            1691000    0310
L0:  END END;                                                               1692000    0310
                                                                  0040 IS 0312 LONG, NEXT SEG 0038
                                                                  0038 IS 0265 LONG, NEXT SEG 0006
PROCEDURE MAINPRO;                                                          1693000    0319
BEGIN                                                                       1694000    0319
COMMENT  THE FOLLOWING PROCEDURES ARE USED:  SRINPUT,SRMAIN,SRANSWER,       1695000    0319
 SRDREAM;                                                                   1696000    0319
LABEL L5;                                                                   1697000    0319
                                                                  START OF SEGMENT ********** 0041
L5: SRINPUT;                                                                1698000    0000
SRMAIN;                                                                     1699000    0000
SRANSWER;                                                                   1700000    0001
SRDREAM;                                                                    1701000    0001
GO TO L5;                                                                   1702000    0002
 END;                                                                       1703000    0002
```

```
                                                                    0041 IS 0003 LONG, NEXT SEG 0006
        COMMENT  INITIALIZING BLOCK;                                        1704000    0319
          XPR=0=K=0;                                                        1705000    0319
        MAINPROC FINISH                                                     1706000    0321
        END;                                                                1707000    0322
                                                                    0006 IS 0325 LONG, NEXT SEG 0002
        LKNJA=(TIME(2)-LKNJA)/60;OKVQK=(TIME(3)-OKVQK)/60;FZOVC=TIME(1);BLZAT/=N    1708000    0056
        IF(PMINT1PAGE1);WRITE(PRINT,CMGUB,'00'=LJLOU=GCPOV,LKNJA,OKVQK);    1709000    0064
        END.                                                                1710000    0083
                                                                    0002 IS 00P6 LONG, NEXT SEG 0001
      EXP IS SEGMENT NUMBER 0042,PRT ADDRESS IS 0101
      LN IS SEGMENT NUMBER 0043,PRT ADDRESS IS 0107
      SQRT IS SEGMENT NUMBER 0044,PRT ADDRESS IS 0583
      OUTPUT(N) IS SEGMENT NUMBER 0045,PRT ADDRESS IS 0044
      OUTPUT(C) IS SEGMENT NUMBER 0046,PRT ADDRESS IS 0041
      INPUT(N) IS SEGMENT NUMBER 0047,PRT ADDRESS IS 0716
      INPUT(C) IS SEGMENT NUMBER 0048,PRT ADDRESS IS 0715
      GO TO SOLVEN  IS SEGMENT NUMBER 0049,PRT ADDRESS IS 0113
      FILE CNTRL(N) IS SEGMENT NUMBER 0050,PRT ADDRESS IS 0014
      FILE CNTRL(C) IS SEGMENT NUMBER 0051,PRT ADDRESS IS 0015
      READ/WRITE IS SEGMENT NUMBER 0052,PRT ADDRESS IS 0016
NUMBER OF ERRORS DETECTED = 000,  COMPILATION TIME = 0180 SECONDS.
PRT SIZE=0467;TOTAL SEGMENT SIZE=05313 WORDS;NUM STORAGE REQ.=06092 WORDS+HDR, SEGS.=0052.
ESTIMATED CORE STORAGE REQUIREMENT  =  10632 WORDS.
```

8.3 ALGOL Listing for ACC

The following is the ALGOL listing of the ACC. Cards 1000 through 43000 were provided by the computing center at Fort Monmouth for file definition and to furnish procedures which calculate some of the basic functions.

```
BEGIN                                                                00001000    0000
                                                                   START OF SEGMENT ********** 0002
FILE IN CARD 0(2,10)$                                                00002000    0000
FILE OUT PUNCH 0(2,10)$                                              00003000    0005
FILE OUT PRINT 0(2,15)$                                              00004000    0010
FILE XXXXXX 2(2,15)$                                                 00005000    0015
FILE TAPE1 2(2,15)$                                                  00006000    0020
FILE TAPE2 2(2,15)$                                                  00007000    0025
FILE TAPE3 2(2,15)$                                                  00008000    0030
FILE TAPE4 2(2,15)$                                                  00009000    0035
FILE TAPE5 2(2,15)$                                                  00010000    0040
FILE TAPE6 2(2,15)$                                                  00011000    0045
FILE TAPE7 2(2,15)$                                                  00012000    0050
FILE TAPE8 2(2,15)$                                                  00013000    0055
FILE TAPE9 2(2,15)$                                                  00014000    0060
FILE TAPE10 2(2,15)$                                                 00015000    0065
FILE TAPE11 2(2,15)$                                                 00016000    0070
FILE TAPE12 2(2,15)$                                                 00017000    0075
FILE TAPE13 2(2,15)$                                                 00018000    0080
FILE TAPE14 2(2,15)$                                                 00019000    0085
FILE TAPE15 2(2,15)$                                                 00020000    0090
FILE TAPE16 2(2,15)$                                                 00021000    0095
SWITCH FILE FILESW=XXXXXX,TAPE1,TAPE2,TAPE3,TAPE4,TAPE5,TAPE6,TAPE7,  00022000    0100
   TAPE8,TAPE9,TAPE10,TAPE11,TAPE12,TAPE13,TAPE14,TAPE15,TAPE16$     00023000    0112
LABEL  FINIS$                                                        00024000    0123
REAL ARRAY  DATA[0:63,0:511]$    COMMENT USED WITH DATA STATEMENTS ONLY$  00025000    0123
REAL Q,XPN$    INTEGER K$                                            00026000    0125
FORMAT F(///////"STOP / PAUSE  NO, "-15), OKTL(2560)$                00027000    0125
                                                                   START OF SEGMENT ********** 0003
                                                                   0003 IS 0017 LONG, NEXT SEG 0002
REAL PROCEDURE INT(ARG1)$          VALUE ARG1$        REAL ARG1$     00028000    0125
   INT=SIGN(ARG1)*ENTIER(ABS(ARG1))$                                 00029000    0125
```

```
REAL PROCEDURE TANH(ARG1)}          VALUE ARG1}          REAL ARG1}                00030000    0133
   TANH←((Q←EXP(ARG1×2))-1)/(Q+1)}                                                00031000    0133
REAL PROCEDURE MAX(ARG1,ARG2)}      VALUE ARG1,ARG2}     REAL ARG1,ARG2}           00032000    0140
   MAX←IF ARG1≥ARG2 THEN ARG1 ELSE ARG2}                                          00033000    0140
REAL PROCEDURE MIN(ARG1,ARG2)}      VALUE ARG1,ARG2}     REAL ARG1,ARG2}           00034000    0145
   MIN←IF ARG1≤ARG2 THEN ARG1 ELSE ARG2}                                          00035000    0145
REAL PROCEDURE DIM(ARG1,ARG2)}      VALUE ARG1,ARG2}     REAL ARG1,ARG2}           00036000    0150
   DIM←MAX(ARG1-ARG2,0)}                                                          00037000    0150
REAL PROCEDURE TSIGN(ARG1,ARG2)}    VALUE ARG1,ARG2}     REAL ARG1,ARG2}           00038000    015A
   TSIGN←SIGN(ARG2)×ABS(ARG1)}                                                    00039000    015A
REAL PROCEDURE LOG(ARG1)}           VALUE ARG1}          REAL ARG1}                00040000    0159
   LOG←LN(ARG1)/2,302585092998}                                                   000A1000    0159
PROCEDURE ERROR(ARG1)}              VALUE ARG1}          REAL ARG1}                00042000    0145
BEGIN WRITE(PRINT,F,ARG1)} GO TO FINIS END}                                       000A3000    0145
PROCEDURE MAINPRO}                                                                000AA000    0175
BEGIN                                                                             000A5000    0175
OWN REAL ARRAY SVCTHETA[0:50],SVALB[0:50],SVF[0:50,0:50],                         000A6000    0175
                                                                  START OF SEGMENT ********** 000A
 SVA[0:50,0:50],SVASUM[0:50],SVBF[0:50,0:50],SVSTHETA[0:50],                      000A7000    0008
 SVCMINF[0:50],SVSMINF[0:50],SVCOSGA[0:50]}                                       000A8000    0019
OWN INTEGER DX1}                                                                  000A9000    0026
OWN INTEGER JNPROB,JKL,JIPROB,JNFOR,JNANGLS,JNREFLT,JNNALB,JJ,JKOUNT,             00050000    0026
 J1,JK,JN,JL,J1L,J1L1,J1L2,J1TEST}                                                00051000    0026
OWN REAL JHSORS,JHO,JRO,JOALR,JARC,JSTER}                                         00052000    0026
FORMAT FL50(16),                                                                  00053000    0026
                                                                  START OF SEGMENT ********** 0005
FL60(315),                                                                        00054000    0026
FL70(3R10.7),                                                                     00055000    0026
FL80(7RA.4),                                                                      00056000    0026
FL110(6R11.6),                                                                    00057000    0026
FL55(" "///X1,"RADIATION RESEARCH ASSOCIATES *ACC* PROBLEM   ",I6),               00058000    0026
FL65(/X15,"SCATTERED LIGHT INTENSITY VERSUS ANGLE AND ALBEDO"),                   00059000    0026
```

```
FL75(X1,"SOURCE HEIGHT = ",S1,E10.3,X2,"DETECTOR COORDINATES   HO=",S1,          00060000   0026
 F10.3,"  RD=",S1,F10.3),                                                        00061000   0026
FL85(X2,"ANGLE",X38,"ALBEDO"),                                                  00062000   0026
FL95(X1,"(COSINE)",A3,R6,A,6R11,4),                                             00063000   0026
FL105(X1,R7,4,X1,S1,7E11,4),                                                    00064000   0026
FL115(/X2,"TOTAL",X2,S1,7E11,4),                                                00065000   0026
FL125(/),                                                                       00066000   0026
FL135(/X10,                                                                     00067000   0026
 "SCATTERED LIGHT CURRENT (PER, PLANE) VERSUS ANGLE AND ALBEDO"),               00068000   0026
FL145(/X10,                                                                     00069000   0026
 "SCATTERED LIGHT CURRENT (HOR, PLANE) VERSUS ANGLE AND ALBEDO");               00070000   0026
                                                                    0005 15 013A LONG, NEXT SEG 0004
LIST LIST1(JNPROB);                                                             00071000   0026
LIST LIST2(JIPROB);                                                             00072000   0031
LIST LIST3(JNCUR);                                                              00073000   0036
LIST LIST4(JNANGLS,JNREFLT,JNNALB);                                             00074000   0041
LIST LIST5(JHSORS,JHO,JRO);                                                     00075000   0049
LIST LIST6(FOR OX1=1 STEP 1 UNTIL JNNALB DO SVALB(OX1));                        00076000   0057
LIST LIST7(JOALH,JARC,JSTER);                                                   00077000   0066
LIST LIST8(FOR OX1=1 STEP 1 UNTIL JNANGLS DO SVCTHETA(OX1));                    00078000   0074
LIST LIST9(FOR OX1=1 STEP 1 UNTIL JNREFLT DO SVFIJJ,OX1));                      00079000   0083
LIST LIST10(FOR OX1=J1L1 STEP 1 UNTIL J1L2 DO SVALB(OX1));                      00080000   0093
LIST LIST11(SVCTHETA(J11),FOR OX1=J1L1 STEP 1 UNTIL J1L2 DO SVB(OX1,            00081000   0102
 J11));                                                                         00082000   0107
LIST LIST12(FOR OX1=J1L1 STEP 1 UNTIL J1L2 DO SVASUM(OX1));                     00083000   0113
LIST LIST13(FOR OX1=J1L1 STEP 1 UNTIL JNNALB DO SVALB10X1));                    00084000   0122
LIST LIST14(SVCTHETA(J11),FOR OX1=J1L1 STEP 1 UNTIL JNNALB DO SVB(OX1,          00085000   0131
 J11));                                                                         00086000   0136
LIST LIST15(FOR OX1=J1L1 STEP 1 UNTIL JNNALB DO SVASUM10X1));                   00087000   0142
LABEL L4A,L4,L3,L8,L9,L10,L11,L13,L14,L15,L16,L19,L20,L22,L23,L2A,               00088000   0151
 L25,L29,L31,L32,L33,L36,L37,L38,L39,LA1;                                       00089000   0151
SWITCH SWGO1=L6,L6,L3,L3;                                                       00090000   0151
```

```
SWITCH SWG02:=L9,L10,L11;                                   00091000  0157
SWITCH SWG03:=L13,L14,L15;                                  00092000  0162
SWITCH SWG04:=L22,L23,L24;                                  00093000  0167
SWITCH SWG05:=L29,L31,L32;                                  00094000  0172
SWITCH SWG06:=L44,L36,L37,L38;                              00095000  0177
SWITCH SWG07:=L39,L44;                                      00096000  0183
SWITCH SWG08:=L41,L41,L44;                                  00097000  0188
SWITCH SWG09:=L39,L41,L44;                                  00098000  0193
READ(CARD,FL50,LIST111FINIS);                               00099000  0198
JKL:=1;                                                     00100000  0204
DO BEGIN                                                    00101000  0204
  READ(CARD,FL50,LIST2)(FINIS);                             00102000  0208
  READ(CARD,FL50,LIST3)(FINIS);                             00103000  0209
  READ(CARD,FL60,LIST4)(FINIS);                             00104000  0214
  READ(CARD,FL70,LIST5)(FINIS);                             00105000  0219
  READ(CARD,FL80,LIST6)(FINIS);                             00106000  022A
  READ(CARD,FL80,LIST7)(FINIS);                             00107000  0229
  READ(CARD,FL80,LIST8)(FINIS);                             00108000  023A
  JJ:=2;                                                    00109000  0239
  DO BEGIN                                                  00110000  0240
    READ(CARD,FL110,LIST9)(FINIS);                          00111000  0240
    END UNTIL (JJ:=(JJ+111)>JNANGLS);                       00112000  0245
  JKOUNT:=1;                                                00113000  0247
  GO TO SWG01[JNCUR];                                       00114000  0248
L3: J1:=1;                                                  00115000  0250
  DO BEGIN                                                  00116000  0251
    SVSTHETAI[J1]:=SQRT(1-(SVCTHETAI[J1]:=2)) END UNTIL (J1:=1[J1:=1])>JNANGLS);  00117000  0251
  J1:=2;                                                    00118000  0256
  DO BEGIN                                                  00119000  0258
    SVCMINI[J1]:=SVCTHETAI[J1-1]*SVCTHETAI[J1]-SVSTHETAI[J1-1]*SVSTHETAI   00120000  0258
     J1];                                                   00121000  0261
    SVSMIN[J1]:=SQRT(1-SVCMINI[J1])/2);                     00122000  0262
```

```
    SVCOSGA[JI1+ABS(JARC*SVSMIN1,II) END UNTIL (JI+(JI+1))>JN+NGLS)          00123000   0286
  LA1 J1+1)                                                                 00124000   0270
  DO BEGIN                                                                  00125000   0271
    JJ+1)                                                                   00126000   0271
    DO BEGIN                                                                00127000   0272
      SVA[JJ,J11+0 END UNTIL 1JJ+(JJ+1))>JNNALR END UNTIL (J1+(JI+1)1>      00128000   0272
   JN+NGLS)                                                                 00129000   027A
  JK+1)                                                                     00130000   0279
  DO BEGIN                                                                  00131000   0279
    J1+2)                                                                   00132000   0279
    DO BEGIN                                                                00133000   0280
      JJ+2)                                                                 00134000   0280
      DO BEGIN                                                              00135000   0281
        JN+JJ-1)                                                            00136000   0281
        GO TO SWGO2[JKOUNT))                                                00137000   0282
        L02 SVA[JK,J11+SVF(J1,JJ1*((SVALR[JK]/JNALR)+JN1+SVA[JK,J1])       00138000   028A
        GO TO LA1                                                           00139000   0294
        L10) SV0[JK,J11+SVF(J1+J,J1*((SVALR[JK)/JNALR1+JN)*(AH+(SVCTHETA1   00140000   0307
J1-1)+,5*(SVCTHETA[J1)-SVCTHETA[J1-1]))+SVA[JK+J11)                         00141000   0314
        GO TO LA1                                                           00142000   0320
        L11) SV01JK+J11+SVF[J1,J,J1*((SVALR[JK)/JNALR1+JN)*SVCOSGA1         00143000   0323
         J11+SV01JK,J11)                                                    00144000   0330
        LA1 END UNTIL (JJ+(JJ+1))>JNREFIT END UNTIL (J1+1J1+1))>JN+NGLS     00145000   0333
    END UNTIL (JK+(JK+1))>JNNALR)                                           00146000   0337
  JK+1)                                                                     00147000   0340
  DO BEGIN                                                                  00148000   0341
    JI+2)                                                                   00149000   0341
    DO BEGIN                                                                00150000   0342
      GO TO SWGO3[JKOUNT))                                                  00151000   0347
      L13) SVA1JK,JI1+SVA(JK,JI1+SVF[JI,11)                                 00152000   0348
      GO TO L16)                                                            00153000   0349
      L'A) SVA1JK,JI1+SVA1JK,JI1+SVF(JI,11*(ABS(SVCTHETA[J1-11+,5*(SVCTH    00154000   0351
```

```
ETA[JI]=SVCTHETA[JI-1]))};                                          00155000   0356
      GO TO L[6};                                                   00156000   0359
      L15: SVA[JK,JI]+SVA[JK,JI[+SVF[,JI,I[=SVCOSGA[JI]};           00157000   0361
      L16: SVB[JK,JI[+SVA[JK,JI[]/(JSTER*(SVCTHETA[JI=I]-SVCTHETAI  00158000   0366
       JI[])};                                                      00159000   0370
      END UNTIL (JI+(JI+1))>JNANGLS END UNTIL (JK+(JK+1)[>JNNALB};  00160000   0372
  JK+1};                                                            00161000   0377
  DO BEGIN                                                          00162000   0377
    SVASUM[JK[+0 END UNTIL (JK+(JK+1))>JNNALB};                     00163000   0377
  JK+1};                                                            00164000   0381
  DO BEGIN                                                          00165000   0382
    JI+1};                                                          00166000   0382
    DO BEGIN                                                        00167000   0382
      SVASUM[JK)+SVA[JK,JI]+SVASUM[JK[ END UNTIL (JI+(JI+[))>,JNANGLS 00168000   0382
     END UNTIL (JK+(JK+1))>JNNALB};                                 00169000   0387
  JL+0};                                                            00170000   0390
  L19: JIL+JL+1};                                                   00171000   0391
  JIL1+7*JL+[};                                                     00172000   0392
  JIL2+7*(JIL[};                                                    00173000   0394
  JITEST+JNNALB-JIL2};                                              00174000   0395
  IF JITEST<0 THEN GO TO L26};                                      00175000   0396
  WRITE(PRINT[PAGE))};                                              00176000   0397
  WRITE(PRINT,FL55,LIST2)};                                         00177000   0401
  GO TO SWGO4[JKOUNT[};                                             00178000   0404
  L22: WR[TE(PRINT,FL65)};                                          00179000   0406
  GO TO L25};                                                       00180000   0410
  L23: WRITE(PRINT,FL145)};                                         00181000   0411
  GO TO L25};                                                       00182000   0414
  L24: WR[TE(PRINT,FL135)};                                         00183000   0415
  L25: WRITE(P.[NT,FL75,LIST5)};                                    00184000   0418
  WRITE(PRINT,FL85)};                                               00185000   0422
  WRITE(PRINT,FL95,LIST10[};                                        00186000   0426
```

```
WRITE(PRINT,FL125);                                    00187000  04
J1:=2;                                                 00188000  0413
DO BEGIN                                               00189000  0414
  WRITE(PRINT,FL105,LIST11);                           00190000  0415
  END UNTIL (J1:=(J1+1))>JNANGLS;                      00191000  0416
WRITE(PRINT,FL115,LIST12);                             00192000  0420
JL:=JL+1;                                              00193000  0424
IF JITES1>7 THEN GO TO L19;                            00194000  0425
J1L1:=J1L2+1;                                          00195000  0426
L26: WRITE(PRINTF'AGE1);                               00196000  0427
     WRITE(PRINT,FL55,LIST2);                          00197000  0431
GO TO SWGOSE(JKOUNT);                                  00198000  0435
L29: WRITE(PRINT,FL65);                                00199000  0437
GO TO L3;                                              00200000  0440
L31: WRITE(PRINT,FL185);                               00201000  0441
GO TO L33;                                             00202000  0444
L32: WRITE(PRINT,FL135);                               00203000  0445
L33: WRITE(PRINT,FL75,LIST5);                          00204000  0448
WRITE(PRINT,FL65);                                     00205000  0472
WRITE(PRINT,FL95,LIST13);                              00206000  0476
WRITE(PRINT,FL125);                                    00207000  0480
II:=2;                                                 00208000  0483
DO BEGIN                                               00209000  0484
  WRITE(PRINT,FL105,LIST14);                           00210000  0488
  END UNTIL (J1:=(JT+1))>JNANGLS;                      00211000  0488
WRITE(PRINT,FL115,LIST15);                             00212000  0490
GO TO SWGO6(JNCUR);                                    00213000  0494
L34: GO TO SWGO7(JKOUNT);                              00214000  0496
L30: JKOUNT:=2;                                        00215000  0498
GO TO L6;                                              00216000  0498
L37: GO TO SWGO8(JKOUNT);                              00217000  0499
L41: JKOUNT:=3;                                        00218000  0502
```

```
            GU TO L6;                                                    00219000    0502
            L38: GO TO SWGO9(JKOUNT);                                    00220000    0503
            LA8: END UNTIL (JKL+(JKL+1))>JNPROR;                         00221000    0506
          ERROR(0);                                                      00222000    0508
           END;                                                          00223000    0509
                                                                  0004 IS 0517 LONG, NEXT SEG 0002
          COMMENT  INITIALIZING BLOCK;                                   00224000    0175
            XPR+0+K+0;                                                   00225000    0175
          MAINPRO: FINI;                                                 00226000    0177
          END.                                                           99999000    0179
                                                                  0002 IS 0182 LONG, NEXT SEG 0001

     EXP IS SEGMENT NUMBER 0006,PRT ADDRESS IS 0061
     LN IS SEGMENT NUMBER 0007,PRT ADDRESS IS 0067
     SQRT IS SEGMENT NUMBER 0008,PRT ADDRESS IS 016A
     OUTPUT(W) IS SEGMENT NUMBER 0009,PRT ADDRESS IS 007A
     OUTPUT(C) IS SEGMENT NUMBER 0010,PRT ADDRESS IS 0071
     INPUT(W) IS SEGMENT NUMBER 0011,PRT ADDRESS IS 0161
     INPUT(C) IS SEGMENT NUMBER 0012,PRT ADDRESS IS 0162
     X TO THE I IS SEGMENT NUMBER 0013,PRT ADDRESS IS 0169
     GO TO SOLVEM  IS SEGMENT NUMBER 0014,PRT ADDRESS IS 0076
     FILE CNTRL(W) IS SEGMENT NUMBER 0015,PRT ADDRESS IS 0014
     FILE CNTRL(C) IS SEGMENT NUMBER 0016,PRT ADDRESS IS 0015
     READ/WRITE IS SEGMENT NUMBER 0017,PRT ADDRESS IS 0016
NUMBER OF ERRORS DETECTED = 000.  COMPILATION TIME = 0025 SECONDS.
PRT SIZE=0118;TOTAL SEGMENT SIZE=00497 WORDS;DRUM STORAGE REQ.=01114 WORDS;NO. SEGS.=0017.
ESTIMATED CORE STORAGE REQUIREMENT  =  08312 WORDS.
```

REFERENCES

1. Collins, D. G. and M. B. Wells, Monte Carlo Codes for Study of Light Transport in the Atmosphere, Volumes I and II, Radiation Research Associates Report ECOM-00240-F, August 1965.

2. Wells, M. B., D. G. Collins and K. Cunningham, Light Transport in the Atmosphere, Volume I: Monte Carlo Studies, Radiation Research Associates Report ECOM-00240-1, Vol. I, September 1966.

3. Cunningham, K., M. B. Wells and D. G. Collins, Light Transport in the Atmosphere, Volume II: Machine Codes for Calculation of Aerosol Scattering and Absorption Coefficients, Radiation Research Associates Report ECOM-00240-1, Vol. II, September 1966.

4. Elterman, L., Atmospheric Attenuation Model, 1964, in the Ultraviolet, Visible and Infrared Regions for Altitudes to 50 KM, Air Force Cambridge Research Laboratories Report AFCRL-64-740, September 1964.

Unclassified
Security Classification

DOCUMENT CONTROL DATA · R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Radiation Research Associates, Inc. 1506 W. Terrell Ave. Fort Worth, Texas 76104	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b GROUP NA

3. REPORT TITLE

Light Transport in the Atmosphere, Vol. III: Utilization Instructions for the LITF Codes

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Annual Report, 1 August 1965 to 31 August 1966

5. AUTHOR(S) (Last name, first name, initial)

Collins, Dave G.
Wells, Michael B.
Cunningham, Kelly

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
September 1966	211	4

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
Contract DA 28-043 AMC-00240(E)	RRA-T63-3
b. PROJECT NO.	
c.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	ECOM 00240-1, Vol. III

10. AVAILABILITY/LIMITATION NOTICES

Distribution of this report is unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Atmospheric Sciences Laboratory U. S. Army Electronics Command Fort Monmouth, New Jersey

13. ABSTRACT

This is the third of three volumes. Volumes I and II contain other aspects of the study: descriptions of the RRA-42 and RRA-45 codes and their applications to the calculation of aerosol attenuation coefficients and the applications of the LITE codes to analysis of experimental data.

The Monte Carlo procedures designated as LITE-I and LITE-II were developed during a previous contract period for use in studying the transport of light through the earth's atmosphere under various environmental conditions. These procedures have been modified to expand their application to a broader range of physical problems. LITE-I treats monochromatic light emitted from a point source, and LITE-II treats monochromatic plane sources of light. The codes have been written in both ALGOL for the Burroughs B-5500 and FORTRAN-IV for other computers. The codes are sufficiently flexible to treat multiple scattering in an atmosphere in which the air density and the aerosol size distribution vary independently and arbitrarily with altitude. Provision for treating ground and cloud reflection with an albedo method is also available in the codes.

A machine procedure, designated as ACC, was developed for use in converting the scattered intensities computed by the LITE codes for a given ground albedo to scattered intensities for problems where only the magnitude of the ground albedo has changed.

Utilization instructions, input data formats, sample problems and the ALGOL listings of ACC and the improved versions of the LITE programs are given to aid those who wish to utilize the codes.

DD FORM 1 JAN 64 1473

Unclassified
Security Classification

Unclassified
Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Utilization Instructions for the LITE Codes Machine Codes Monte Carlo Methods Light Transmission Radiation Transport Variable Air Density Albedo Point Source Plane Source Multiple Scattering Aerosol Scattering Rayleigh Scattering						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2a. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2b. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7a. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8a. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, & 8d. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9a. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through _____."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through _____."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through _____."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS), (S), (C), or (U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, rules, and weights is optional.

Unclassified
Security Classification